150 ptas.



CURSO PRACTICO DEL ORDENADOR PERSONAL, EL MICRO Y EL MINIORDENADOR

Editorial Delta, S.A.

MI COMPUTER CURSO PRACTICO
DEL ORDENADOR PERSONAL, EL MICRO Y EL MINIORDENADOR

Publicado por Editorial Delta, S.A., Barcelona, y comercializado en exclusiva por Distribuidora Olimpia, S.A., Barcelona

Volumen I - Fascículo 12

Director:	José Mas Godayol
Director editorial:	Gerardo Romero
Jefe de redacción:	Pablo Parra
Coordinación editorial:	Jaime Mardones
Asesor técnico:	Roberto Quiroga

Redactores y colaboradores: G. Jefferson, R. Ford, S. Tarditti, A. Cuevas
Para la edición inglesa: R. Pawson (editor), D. Tebbutt (consultant editor), C. Cooper (executive editor), D. Whelan (art editor), Bunch Partworks Ltd. (proyecto y realización)

Realización gráfica: Luis F. Balaguer

Redacción y administración:
Paseo de Gracia, 88, 5.° - Barcelona-8
Tels. (93) 215 10 32 / (93) 215 10 50 - Télex 97848 EDLTE

MI COMPUTER, *Curso práctico del ordenador personal, el micro y el miniordenador,* se publica en forma de 96 fascículos de aparición semanal, encuadernables en ocho volúmenes. Cada fascículo consta de 20 páginas interiores y sus correspondientes cubiertas. Con el fascículo que completa cada uno de los volúmenes, se ponen a la venta las tapas para su encuadernación.
El editor se reserva el derecho de modificar el precio de venta del fascículo en el transcurso de la obra, si las circunstancias del mercado así lo exigieran.

ISBN: 84-85822-83-8 (fascículo) 84-85822-84-6 (tomo 1)
84-85822-82-X (obra completa)
Depósito Legal: B. 52-84

Fotocomposición: Tecfa, S.A., Pedro IV, 160, Barcelona-5
Impresión: Cayfosa, Santa Perpètua de Mogoda (Barcelona) 048404
Impreso en España - Printed in Spain - Marzo 1984

Editorial Delta, S.A., garantiza la publicación de todos los fascículos que componen esta obra.

Distribuye para España: Marco Ibérica, Distribución de Ediciones, S.A., Carretera de Irún, km 13,350. Variante de Fuencarral, Madrid-34.
Distribuye para Argentina: Viscontea Distribuidora, S.C.A., La Rioja 1134/56, Buenos Aires.
Distribuye para Colombia: Distribuidoras Unidas, Ltda., Transversal 93, n.° 52-03, Bogotá D.E.
Distribuye para México: Distribuidora Intermex, S.A., Lucio Blanco, n.° 435, Col. San Juan Tlihuaca, Azcapotzalco, 02400, México D.F.
Distribuye para Venezuela: Distribuidora Continental, S.A., Ferrenquín a Cruz de Candelaria, 178, Caracas, y todas sus sucursales en el interior del país.

Pida a su proveedor habitual que le reserve un ejemplar de MI COMPUTER. Comprando su fascículo todas las semanas y en el mismo quiosco o librería, Vd. conseguirá un servicio más rápido, pues nos permite realizar la distribución a los puntos de venta con la mayor precisión.

Servicio de suscripciones y atrasados (sólo para España)

Las condiciones de suscripción a la obra completa (96 fascículos más las tapas, guardas y transferibles para la confección de los 8 volúmenes) son las siguientes:

a) Un pago único anticipado de 16 690 ptas. o bien 8 pagos trimestrales anticipados y consecutivos de 2 087 ptas. (sin gastos de envío).

b) Los pagos pueden hacerse efectivos mediante ingreso en la cuenta 3371872 de la Caja Postal de Ahorros y remitiendo a continuación el resguardo o su fotocopia a Distribuidora Olimpia (Paseo de Gracia, 88, 5.°, Barcelona-8), o también con talón bancario remitido a la misma dirección.

c) Se realizará un envío cada 12 semanas, compuesto de 12 fascículos y las tapas para encuadernarlos.

Los fascículos atrasados pueden adquirirse en el quiosco o librería habitual. También pueden recibirse por correo, con incremento del coste de envío, haciendo llegar su importe a Distribuidora Olimpia, en la forma establecida en el apartado b).

Para cualquier aclaración, telefonear al (93) 215 75 21.

No se efectúan envíos contra reembolso.



Contra la máquina

Algunos de los microordenadores más potentes que existen hoy no están sólo en las oficinas o las fábricas de alta tecnología, sino también en las salas recreativas, los bares y las cafeterías



Marcus Wilson-Smith

Invasión extraterrestre
"Tienes que eliminarlo antes de que llegue al punto de explosión atómica, o se reproducirá clónicamente... ¡Cuidado! La nave madre se está preparando para sacar su cápsula... Será mejor que uses la Bomba Inteligente..."
No, no se trata de un diálogo perteneciente a una película de ciencia-ficción, sino de una conversación oída en una sala de juegos recreativos

En 1971, un joven llamado Nolan Bushnell invirtió gran cantidad de tiempo y de energía tratando de convencer a los propietarios de los bares y las cafeterías de los alrededores de Sunnyvale (California), donde vivía, para que accedieran a probar un nuevo tipo de juego que él mismo había inventado. Se trataba de una máquina tragamonedas, pero respondía a un concepto totalmente nuevo que ahora conocemos como televisión interactiva.

Finalmente el propietario de un bar accedió a probarla. Dos días después le telefoneó a Bushnell, quejándose porque la máquina se había averiado. Cuando Bushnell llegó al bar, descubrió que no había avería: la caja de las monedas estaba repleta.

El *Pong* (un derivado del tenis de mesa o ping-pong) fue el precursor de todos los emocionantes e innovadores juegos recreativos basados en microordenadores que ahora se pueden hallar en cualquier rincón del mundo. Tal vez sea interesante anotar que, si bien Bushnell fue el primero en sugerir que juegos como el tenis de mesa se podían simular mediante un ordenador, aun así no había llegado aún a formular el avance innovador decisivo: su juego exigía dos jugadores que compitieran entre sí, en vez de que fuera un solo jugador el que empleara toda su capacidad e inteligencia contra la máquina. Hubo de transcurrir un tiempo sorprendentemente largo hasta que surgiera la siguiente generación de juegos recreativos. No fue hasta 1977 que una empresa japonesa denominada Taito apareció en escena con el *Space invaders* (Invasores del espacio), que obtuvo un éxito enorme.

En la terminología que se emplea actualmente, *Space invaders* se conoce como un juego *alien zapping with shields* (eliminar extraterrestres, protegiéndose con escudos). El jugador desplaza su estación de disparo a lo largo de la parte inferior de la pantalla, protegiéndose con escudos cada vez que se siente amenazado, y disparando contra una línea de seres extraterrestres estilizados que siempre avanzan hacia él y que también le disparan a intervalos irregulares. El ritmo del avance de los extraterrestres es totalmente predecible y se acompaña con un ruido eléctrico apropiado en dos tonos que marca el paso de la desaceleración.

El *Space invaders* original utilizaba una visualización de televisión monocromática y gráficos de barrido. Era muy poco innovador en términos tanto de software como de hardware, pero cuando apareció por primera vez provocó una revolución social cuyo

precedente más cercano se remonta al nacimiento del cine. El hardware que soportaba al *Space invaders* y sus congéneres más cercanos apenas se diferenciaba del de los micros personales de entonces. Los fabricantes de estos ordenadores comprendieron rápidamente que existía un gran mercado aún por explotar en los prósperos hogares de Occidente. Las campañas de comercialización pasaron con prontitud de dirigir su enfoque hacia el aprendizaje de lenguajes para escribir programas para ordenador a centrarse en la utilización de éste como un medio de entretenimiento, y el diseño de las máquinas siguió el mismo rumbo.

Siete años después, los ordenadores aún se siguen vendiendo en función de su potencial como fuente de juegos, si bien actualmente la última generación de juegos recreativos ha ido mucho más allá, equiparando su capacidad a la de los ordenadores personales más avanzados. Es bastante frecuente encontrarse con un juego recreativo que tenga hasta un millón de bytes de memoria, así como una capacidad para gráficos muy raros de encontrar como no sea en los terminales para gráficos construidos especialmente para que los utilicen, con ordenadores de unidad principal, diseñadores, arquitectos, etc.



Muchos de los adelantos que se han producido en el campo de la informática personal se remontan a los juegos recreativos. El paso hacia los procesadores de 16 e incluso de 32 bits, por ejemplo, se produjo a consecuencia de los requerimientos de los usuarios de direccionar más de 64 Kbytes y de su necesidad de mayores velocidades de procesamiento. Los fabricantes de juegos recreativos estuvieron en la vanguardia de este movimiento y fueron unos de los primeros clientes de procesadores de 16 bits como el 68000 de Motorola y el 8086 de Intel. Sus exigencias en cuanto a un procesamiento más rápido y mayores memorias precedieron incluso a aquellas de los usuarios de microordenadores de oficina.

Todo ello ha sido de gran beneficio para el usuario de ordenadores personales, incluso de los del extremo inferior de la escala. Los gráficos sprite, por ejemplo, se desarrollaron para los juegos recreativos y posteriormente comenzaron a estar disponibles para los ordenadores personales. Chips especializados para gráficos y generación de sonido, como el Video Interface Chip y el Sound Interface Device, de Commodore, surgieron todos de la misma fuente, al igual que los tres chips que empleara Atari con la misma finalidad en sus ordenadores personales de las series 400 y 800.

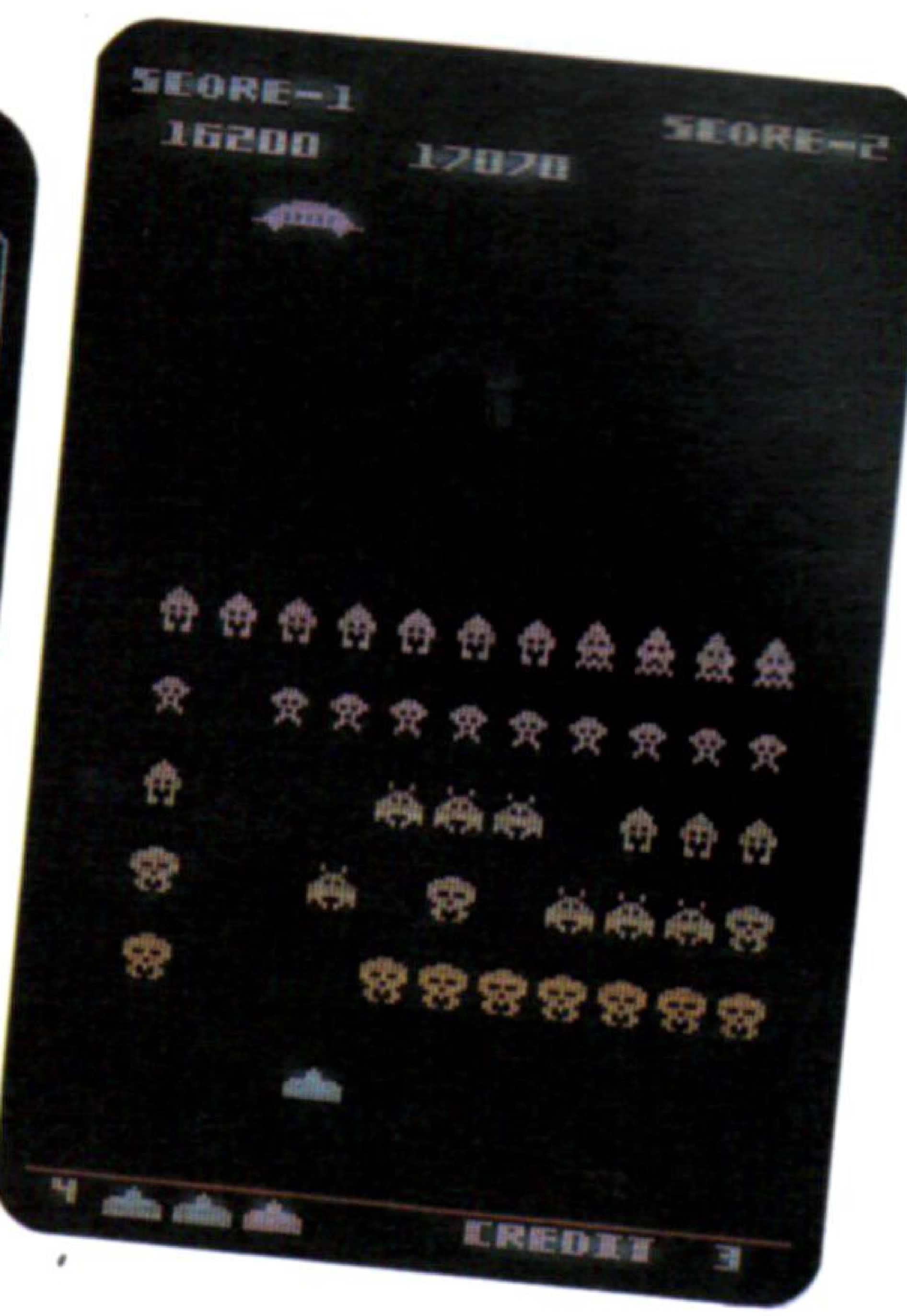
Atari, cuyos éxitos se deben todos al juego original de Nolan Bushnell, constituye un ejemplo particularmente adecuado de la interrelación entre los juegos recreativos y los ordenadores personales. Hace mucho tiempo que Atari viene considerando el ordenador personal antes que nada como un medio de entretenimiento (un reflejo, quizá, del interés primordial de su propietario, ya que Atari pertenece a la Warner Bros) y, además de su gama de micros, también ofrece un ordenador exclusivo para juegos, el VCS (Video Cartridge System), que introduce en el hogar, virtualmente sin ninguna modificación, muchos de los juegos existentes para máquinas recreativas. En este sentido Atari tiene, por supuesto, la gran ventaja de que es una de las principales productoras de máquinas recreativas. Sin embargo, otras empresas dedicadas a cubrir el campo del entretenimiento y el ocio, en particular CBS (Columbia Broadcasting System) y la firma fabricante de juguetes Mattel, están también muy comprometidas en el mismo negocio.

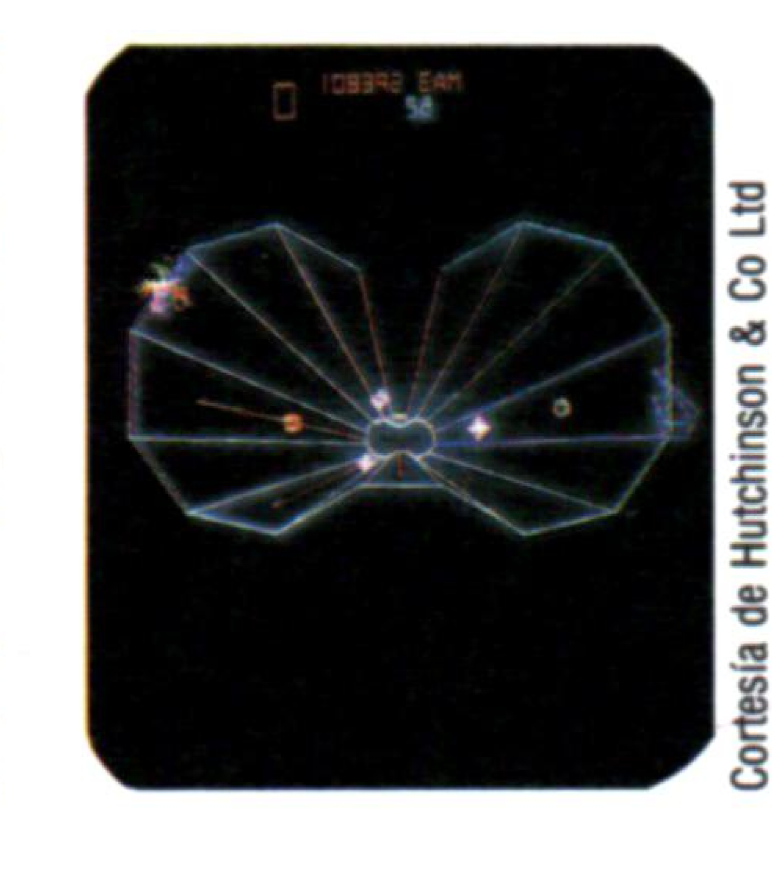
Todo el mercado del entretenimiento, trátese de juegos recreativos unitarios como de software de juegos para micros personales, se está convirtiendo rápidamente en toda una industria en sí mismo. En términos de estrategia de marketing guarda mucha semejanza con la industria discográfica, al proporcionar regularmente listas que incluyen los 20 juegos más vendidos. Ambos fenómenos ofrecen, asimismo, otros puntos de similitud. El software de juegos parece ser una de las obsesiones actuales de los niños en edad escolar y existe un floreciente negocio de piratería de software (véase p. 192).

Hemos mencionado antes que el nivel de desarrollo del hardware que se puede encontrar en los juegos recreativos los convierte en una clase diferente respecto a los micros personales. El más reciente de estos adelantos técnicos introduce la utilización de videodiscos para proporcionar, en el monitor de televisión, un telón de fondo sobre el cual se realizan los juegos. Si refinamientos como éste llegarán algún día al mercado personal, es tema de pura especulación. Ciertamente, existe en el campo de la electrónica tecnología avanzada a precio razonable para el consumidor, aunque el software de juegos que se produce para el mercado personal, tanto para consolas exclusivas para juegos como para micros personales, no ha llegado al nivel de los juegos recreativos en su forma original.

Cuando hablábamos de los simuladores de vuelo (véase p. 201), señalábamos que todos los juegos recreativos son simulacros de una u otra clase, ya sea de







Campos de juego
Los juegos recreativos como *Space invaders*, en los cuales el jugador puede mover su "señal" sólo a lo largo de una línea fija, han sido superados en popularidad por los juegos de "persecución en laberintos" como *Pac-Man*. Ambos utilizaban gráficos sprite y, en consecuencia, su calidad visual era pobre. Recientemente, los diseñadores de juegos han estado produciendo representaciones mucho más abstractas, como *Battlezone*, en que un jugador futurista combate contra un juego de tanques y misiles, o *Tempest*, que utiliza un asombroso diseño de gráficos para producir una magnífica ilusión de profundidad

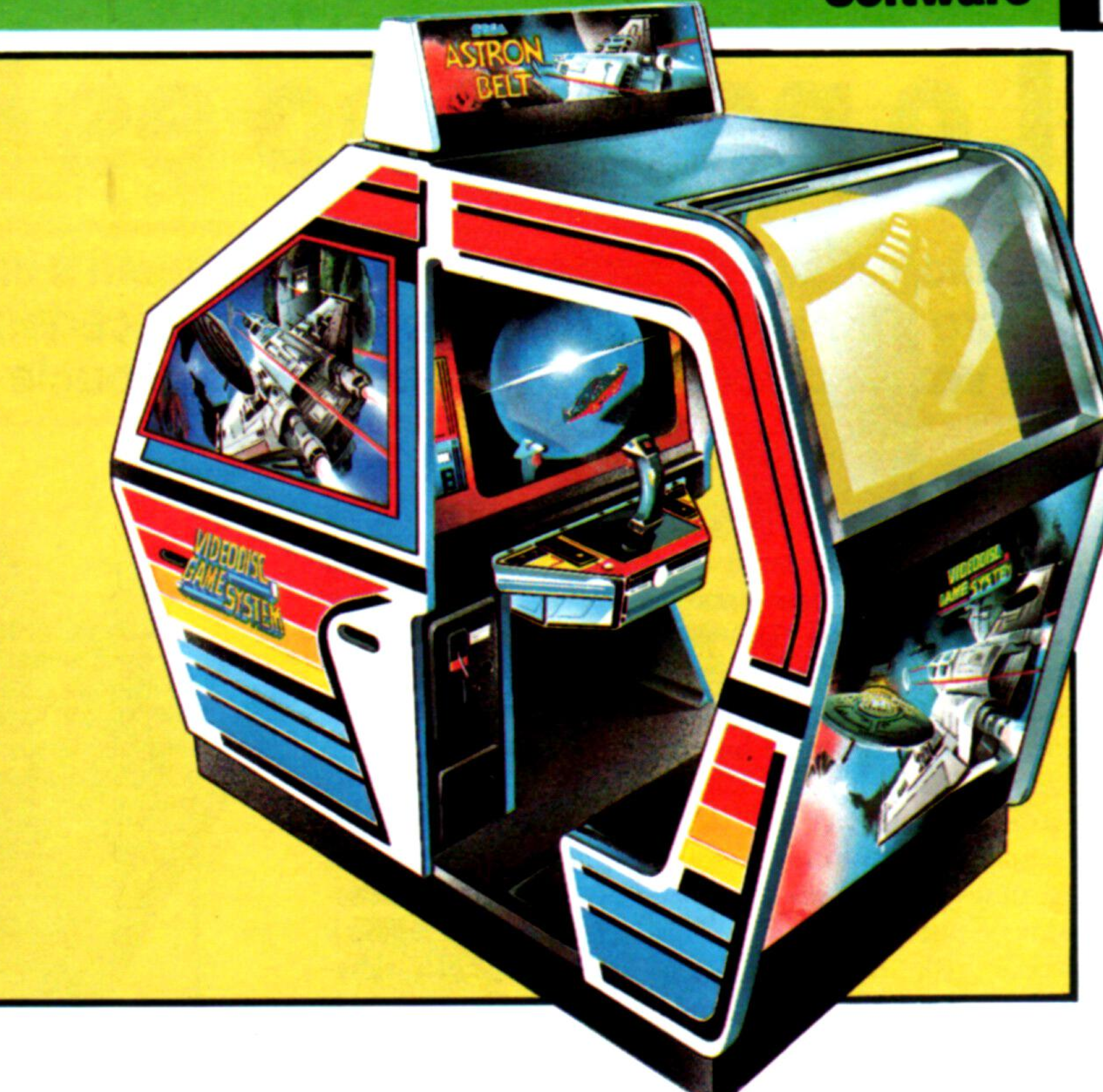
Cortesía de Hutchinson & Co Ltd

Cápsula espacial

Astron belt es un ejemplo típico de la nueva generación de juegos recreativos que utilizan discos láser (véase abajo) para proporcionarle al juego un fondo en movimiento. Como los discos son dispositivos de acceso al azar, se puede pasar directamente de una escena a otra: una confrontación contra una nave extraterrestre que concluye con una explosión, por ejemplo. Para el jugador esto representa un gran paso hacia un mayor realismo. Las representaciones del disco láser pueden ser imágenes en movimiento de la vida real o bien imágenes animadas generadas por ordenador. El *Astron belt* también ofrece sonido estereofónico e incluso un asiento vibrador que reacciona a las bajas frecuencias del circuito de audio



Tony Lodge



situaciones de la vida real, como un juego de tenis de mesa o una carrera de coches, o bien una fantasía, como *Space invaders, Pac-Man* o *Frogger*. En los doce años transcurridos desde que existen los juegos recreativos, estas dos amplias bases se han ampliado, convirtiéndose en corrientes de desarrollo paralelas, si bien los juegos de fantasías espaciales son más comunes.

Hemos hablado anteriormente de los dos juegos originales (el *Pong* y *Space invaders*), pero vale la pena referirnos a su desarrollo. El primero de los juegos de bate y pelota que enfrentó al hombre contra la máquina fue el *Breakout* y todas sus variantes, en el cual el jugador arroja una pelota contra una pared de ladrillos. Cada ladrillo al ser golpeado desaparece de la pantalla, y el objetivo consiste en hacerlos desaparecer todos sin perder la pelota. A partir de este esquema tenemos los "simulacros" de golf, snooker-pool y pinball. En líneas generales, cuanta más atención preste el programador a reproducir las fuerzas existentes en el mundo real (por ejemplo, la gravedad, la resistencia del viento o de la tierra, un golpe impreciso), mejor será el juego.

Pero estos criterios no se aplican a los juegos de fantasía. En este caso, el jugador en realidad compite contra la persona que programó la máquina para que jugara al juego en cuestión, actuando enteramente según *sus* reglas, en un universo de *su* creación. Después de *Space invaders*, la etapa siguiente fue incluir distintas clases de extraterrestres que atacaran de diferente manera a intervalos al azar. Después de esto, fue necesario introducir movilidad en la "señal de pantalla" del jugador, lo que llevó a la creación de juegos tales como el *Defender*, considerado por los entendidos como el mejor de todos los de este tipo.

Los juegos de línea fija, como el *Space invaders* original, siguieron evolucionando y se convirtieron en lo que es el *Missile command* y sus derivados, cuyo objetivo consiste en defender la base propia contra el ataque de misiles intercontinentales. Los juegos de "persecución en laberintos", como *Pac-Man*, se desarrollaron a partir de los juegos de carreras de coches originales, donde el fin último consistía en conducir una señal en forma de coche por un recorrido la mayor cantidad de veces posible en el tiempo permitido, sin chocar contra las paredes. En estos juegos no había un verdadero elemento de competición (ni siquiera considerando las manchas de aceite que aparecían al azar como por arte de magia), de manera que el paso siguiente fue el convertir la "prueba de tiempo" en una persecución. La pista de coches original se convirtió entonces en un laberinto y las señales, por su parte, fueron sustituidas por frutas, bulbos luminosos y cosas por el estilo.

Los juegos de persecuciones de coches se orientaron hacia una representación seudotridimensional del recorrido, contemplado desde el interior del coche o desde detrás de él, con la carretera, siempre cambiante, abalanzándose sobre el jugador. Un método muy similar es el que se utiliza en los juegos de simulación de vuelo más realistas.

Por último, están los juegos tradicionales de tablero, como las damas, el ajedrez y el backgammon. Éstos están confinados a las aplicaciones para ordenadores personales, porque normalmente para jugar a ellos se necesita mucho más tiempo y la representación de gráficos ocupa un segundo lugar en importancia, después de los propios algoritmos de juegos del programa.

La única diferencia técnica verdadera radica en el método utilizado para generar los gráficos. La totalidad de los juegos más antiguos, así como la mayor parte de los actuales, emplean gráficos de barrido; pero algunos, en particular el *Asteroids*, utilizan métodos de exploración por vector, por lo cual el haz de electrones explora sólo las imágenes de la pantalla (no las zonas oscuras).

De modo que la próxima vez que pase por una sala recreativa repleta de juegos sumamente complicados, o que se apoye contra una de estas máquinas en un bar, recuerde que, en su interior, el ordenador que las acciona probablemente es mucho más complejo y poderoso que cualquiera de los que se emplean en el hogar o en una pequeña oficina, que utiliza muchas de las mismas técnicas y que su software ha sido creado por algunos de los mejores programadores que existen en la actualidad en todo el mundo.

Lo pequeño es bello

El microdisco de Sinclair es una respuesta a los problemas de costo y dimensiones que plantea el almacenamiento de datos. En vez de disco flexible usa un estrecho bucle de cinta magnética

Para el usuario de ordenadores personales, la cassette convencional representa un método barato y, por lo general, fiable para almacenar programas o cargar software comercial. Sin embargo, el sistema de cassette posee diversos inconvenientes. El problema principal es el que atañe a la velocidad; incluso un sistema de cassette rápido que funcione a 1 200 bits por segundo puede tardar varios minutos en cargar un programa largo o en buscar una información determinada. El segundo problema relevante se refiere a la cinta, que se mueve en un solo sentido: por lo general el ordenador no puede controlar los mandos de avance rápido y rebobinado. En el caso de que el programa esté almacenado en el extremo de una cinta, antes de empezar a cargarlo, será preciso bobinar la cinta entera a través de la grabadora.

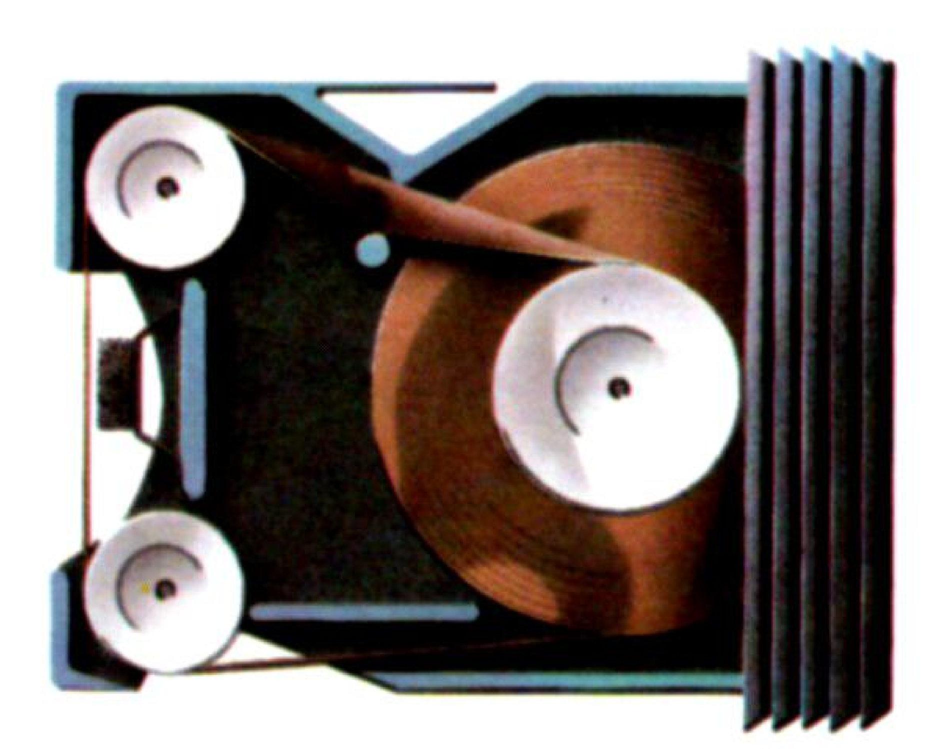
Un sistema de disco resuelve todos estos inconvenientes, pero su precio es poco asequible. Lo que en realidad necesitan muchos usuarios de ordenadores

Microdisco cortesía de Sinclair Research

personales es un dispositivo que sea mucho más veloz que una cassette y, a la vez, mucho más económico que un disco. Estos sistemas existen y se denominan *floppy tapes* (cintas flexibles). Desarrollada originalmente en Estados Unidos por Exactron para el sistema TRS-80 Modelo 1 de Tandy, la primera cinta flexible utilizaba un bucle de cinta continuo alojado en el interior de un cartucho; la idea surgió del sistema de cinta de audio de ocho pistas que estaba de moda hace algunos años. El princpio de funcionamiento es sencillo: el bucle de cinta circula constantemente, de manera que los distintos programas se pueden hallar con muchísima más rapidez. También se lleva un catálogo de todos los programas y los archivos almacenados en la cinta (como el directorio de un disco), de manera que el usuario siempre tiene a su disposición una lista de contenido.



Cable para interface
Este cable plano flexible conecta el primer microdisco con la unidad de interface



El wafer del microdisco
La cinta alojada en el interior del wafer del microdisco es un bucle de cinta de video continua de alrededor de 2 mm de ancho. Se utiliza cinta de video en vez de cinta de audio simplemente por su resistencia y su prolongada vida. Comparada con una cassette de audio, la cinta es más delgada y más estrecha, por lo cual es sumamente frágil.
En funcionamiento, el bucle de cinta circula dentro del microdisco en aproximadamente siete segundos y la información se transfiere a alrededor de seis Kbytes por segundo, un significativo avance respecto al 1,5 Kbyte por segundo de la interface para cassette del Spectrum. Cualquier programa almacenado en el wafer se puede hallar y cargar en aproximadamente 15 segundos. Cada cinta puede retener hasta 100 Kbytes de información, pero Sinclair sólo garantiza 86 Kbytes útiles. Cada wafer se ha de preparar para su utilización "dando formato" a la cinta, proceso que se activa mediante una sencilla orden en BASIC. A través del formato se verifica qué partes de la cinta se pueden emplear y se saltan los trozos en malas condiciones

Como la información se graba digitalmente y no por métodos de audio, la transferencia de datos se puede efectuar con mucha mayor celeridad que con una cinta de cassette de audio: al menos cinco veces más rápido y, en ocasiones, aún más. No se requiere ninguna interface compleja ni cara: la unidad utiliza una conexión normal en paralelo y todo el software operativo necesario se encuentra incorporado en la unidad o bien se presenta como una ROM que se enchufa en un conector de reserva en el interior del ordenador. El nombre con que se conocen estas unidades describe su condición, a medio camino entre una cinta y un disco flexible: utilizan cinta pero funcionan como un disco.

Lamentablemente, las primeras unidades estuvieron plagadas de problemas. Los mecanismos funcionaban bien, pero las cintas resultaban poco fiables. El mayor fallo consistía en que la cinta, en realidad, no resistía la tensión de ser constantemente estirada desde una bobina y vuelta a bobinar en su exterior. Este problema no se presentaba en las cintas de audio de ocho pistas, que eran mucho más anchas y se movían mucho más lentamente. Hasta la introducción del microdisco de Sinclair, no es exagerado decir que el sistema de cinta flexible se consideraba algo así como una causa perdida.

El microdisco funciona exactamente según los mismos principios: un bucle de cinta pasa de manera constante a través de una cabeza de grabación y reproducción. Más aún: la cinta jamás había sido tan pequeña, con una anchura de alrededor de 2 mm (menos de la mitad que la cinta flexible original). Sin embargo, su fiabilidad está aún por demostrarse.

La cinta de cassette digital constituye un sistema de almacenamiento alternativo muy veloz y de probada fiabilidad. Hace ya mucho que las unidades profesionales están a la venta, pero a precios muy elevados. Con la introducción del sistema de microcassette digital de Philips han comenzado a aparecer dispositivos como la Hobbit (véase p. 94). A pesar de que la cinta no está en forma de bucle, ofrece una notable velocidad. El directorio se encuentra en la mitad de la cinta, que se puede bobinar en ambos sentidos bajo el control del sistema operativo. Recientemente se han incorporado aplicaciones de este tipo de microcassettes en los ordenadores portátiles Sharp PC-1251 y Epson HX-20.

El único problema importante de estos tres sistemas de almacenamiento es que el software disponible en sus respectivos formatos es limitado. Los ordenadores PC-1251 y HX-20 disponen de medios para cargar programas desde una cassette convencional y guardarlos después en la microcassette interior. La Hobbit y el microdisco se pueden conectar al ordenador simultáneamente como si se tratara de una grabadora de cassettes corriente, con lo cual se simplifica aún más el traspaso de información.

En términos de fiabilidad demostrada, el sistema de microcassette digital les lleva ventaja tanto al de cinta flexible como al de cassette convencional. La posibilidad de que con el tiempo llegue a reemplazarlos, entra en el campo de la especulación.

Sin rodeos

¿Qué ordenador adquirir y de qué modo mejorar su rendimiento con accesorios y periféricos? He aquí un informe objetivo sobre el hardware existente

Periféricos
Cuando se anuncia un ordenador nuevo, frecuentemente el folleto describe una gama de periféricos que tal vez no están ni siquiera en fase de diseño. Por lo tanto, es importante distinguir entre lo que se encuentra a la venta y lo que sólo está planificado para el futuro. El ordenador para el cual existan tanto unidad de disco como impresoras producidas por el mismo fabricante es una máquina para la cual se desarrollarán útiles programas de oficina. Para fines de entretenimiento, una máquina bien planificada habrá de contar con palancas de mando, mandos de raqueta y lápices ópticos disponibles, aunque en estos casos con frecuencia se puede sacar mejor partido de los accesorios suministrados por los proveedores independientes

> > > Spectrum, BBC Modelo B, Tandy Color, TI99/4A
> > Atari 400 y 800, Vic-20, Commodore 64, Dragon-32
> Oric-1, Sord M5

Pantalla
A los gráficos del ordenador se les presta mucha atención, cuando de hecho la mayor parte del tiempo es probable que el usuario esté mirando texto o listados de programas. Dos factores son importantes: el número de caracteres que se pueden visualizar simultáneamente (la cifra promedio es de 25 líneas de 40 caracteres) y la legibilidad de los caracteres. También son importantes las configuraciones de edición en pantalla. ¿Se pueden introducir alteraciones en un programa simplemente moviendo el cursor hasta la posición y digitando, o se deben utilizar órdenes especiales?

> > > BBC Modelo B, Atari 400 y 800, Commodore 64
> > Oric-1, Sord M5, TI99/4A
> Spectrum, Vic-20, Dragon-32, Tandy Color



Teclado
A menos que sólo pretenda jugar empleando una palanca de mando, utilizará su teclado con mucha frecuencia. Por lo tanto es importante que se sienta a gusto con él. La calidad de un teclado viene determinada, en primer lugar, por el diseño físico o el "tacto". Un mecanógrafo al tacto, por ejemplo, probablemente preferirá un teclado de los de mayor calidad, ya que algunos de los ordenadores más baratos están equipados con teclas transistorizadas que no ofrecen "feedback táctil". En segundo lugar está el trazado del teclado. El ideal es aquel que posee una tecla individual para cada función, ya que el hecho de que una misma tecla posea cierta cantidad de posibles aplicaciones diferentes puede llevar a confusión. Vale la pena disponer de teclas de función programable, porque se pueden utilizar en un programa para realizar funciones especiales como FIRE o START AGAIN

> > > BBC Modelo B, Atari 800, Vic-20, Commodore 64
> > Dragon-32, Tandy Color, TI99/4A
> Spectrum, Atari 400, Oric-1, Sord M5



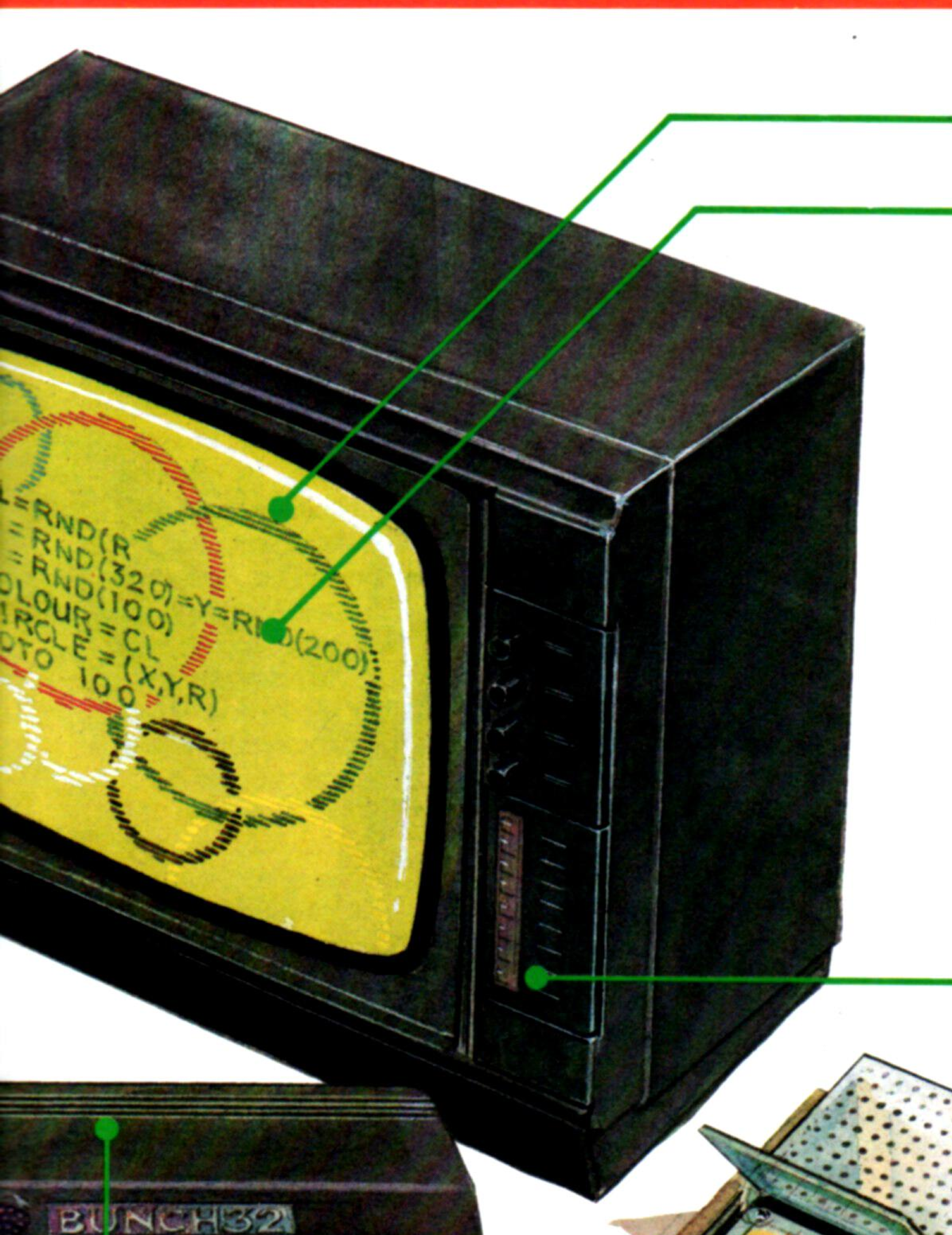
Programas de oficina
Si su razón principal para la adquisición de un microordenador reside en la capacidad de estas máquinas para llevar una pequeña empresa, debería pensar en un ordenador diseñado con ese fin. Sin embargo, los ordenadores personales más caros también se pueden utilizar como máquinas contables siempre y cuando esté dispuesto a invertir en una impresora y una unidad de disco. En ese caso las aplicaciones comprenden tratamiento de textos, contabilidad, bases de datos y hojas electrónicas.

> > > Atari 800, Commodore 64
> > BBC Modelo B, Atari 400, Vic-20, TI99/4A
> Spectrum, Dragon-32, Oric-1, Sord M5, Tandy Color

Interfaces
Si un ordenador posee una amplia gama de interfaces, es que ha sido diseñado previendo una futura ampliación, y esta característica constituye un excelente indicio de buen diseño en general. Disponer de conexiones estandarizadas como Centronics y RS232 significa poder utilizar una amplia gama de dispositivos para conectar a su ordenador, aparte de las unidades proporcionadas por el fabricante

> > > BBC Modelo B, Vic-20, Commodore 64
> > Atari 400 y 800, Dragon-32, Oric-1, Sord M5, Tandy Color
> Spectrum, TI99/4A





Gráficos
Es difícil comparar la calidad de los gráficos que presentan los diversos ordenadores personales, ya que la mayoría de ellos ofrecen más de una modalidad. Digitar MODE 1, por ejemplo, podría darle acceso a 16 colores en una cuadrícula de 40 × 25 posiciones, mientras que MODE 7 podría proporcionar una resolución máxima de 320 × 200 pixels, pero con una elección de tan sólo dos colores. Debe tener en cuenta que en algunas máquinas la máxima resolución únicamente se puede apreciar si utiliza como pantalla un monitor en lugar de un televisor

> > > BBC Modelo B, Atari 400 y 800, Commodore 64, Sord M5, TI99/4A
> > Spectrum, Dragon-32, Oric-1, Tandy Color
> Vic-20

La calidad del BASIC
Casi todos los ordenadores personales vienen con un intérprete de BASIC incorporado en la ROM, pero como habrá observado a partir de nuestros recuadros de "Complementos al BASIC", las órdenes disponibles en cada uno de ellos varían considerablemente. Todas las versiones de BASIC que ostentan el nombre Microsoft trabajan aproximadamente de la misma manera, otras pueden utilizar estructuras completamente diferentes para series, matrices, etc. Un buen BASIC es aquel que posee muchas órdenes de "alto nivel", tales como CIRCLE, DRAW y PAINT

> > > BBC Modelo B, Dragon-32, Spectrum, Atari 400 y 800, Oric-1, Tandy Color, TI99/4A
> > Vic-20, Commodore 64
> Sord M5

Sonido
Los ordenadores que utilizan el altavoz del televisor por lo general pueden producir un sonido de mayor calidad que los ordenadores que emplean un altavoz incorporado, aunque esto no es válido en todos los casos; de manera que es conveniente que antes de comprar un ordenador se solicite una demostración. Un sistema de voz única permite sólo notas únicas (como tocar el piano con un dedo), y tres o cuatro voces pueden realizar acordes. Los generadores de "ruido blanco" que poseen algunos ordenadores se utilizan para crear explosiones y otros efectos, mientras que aquellos con "regulador de envoltura" pueden imitar distintos instrumentos musicales, como un minisintetizador

> > > BBC Modelo B, Atari 400 y 800, Commodore 64, Oric-1, Sord M5, TI99/4A
> > Vic-20
> Spectrum, Dragon-32

Memoria
Cuanta más RAM posee un ordenador, más sofisticados son los programas que puede ejecutar. Los ordenadores personales vienen con una memoria estándar que varía entre 1 y 64 K, susceptible de ampliación en la mayor parte de ellos. No obstante, el ordenador puede muy bien requerir parte de esta memoria para su propio uso interno (denominado *overheads* del sistema), dejando menor cantidad para el programa. La visualización de gráficos en color de alta resolución consume grandes cantidades de RAM

> > > Spectrum (48 K), BBC Modelo B, Atari 800, Commodore 64, Dragon-32, Oric-1 (48 K)
> > Atari 400, Tandy Color (16 K), TI99/4A
> Vic-20, Sord M5 (4 K)

Grabadora de cassette
A menos que esté dispuesto a ceñirse exclusivamente al cartucho de ROM o al software basado en disco, el usuario necesitará una grabadora de cassette. Muy pocos fabricantes exigen todavía que se compre su propia unidad; la mayoría de los ordenadores funcionan con cualquier grabadora de cassette doméstica, si bien tanto la velocidad como la fiabilidad con que se guardan y recuperan los programas varían mucho. Los mejores sistemas permiten controlar también el motor de la grabadora de cassette

> > > Spectrum, BBC Modelo B
> > Dragon-32, Oric-1, Sord M5, Tandy Color, TI99/4A
> Atari 400 y 800, Vic-20, Commodore 64

Software de juegos
Gran parte de los mejores programas para juegos existentes no los vende el propio fabricante del ordenador sino proveedores especializados en software. Lo que determina tanto la calidad como la cantidad de los juegos producidos para cualquier máquina es el número de usuarios de ésta (y, en consecuencia, el volumen de ventas potenciales) y sus especificaciones técnicas. Por este motivo, algunos ordenadores parecen destinados a juegos mientras otros se orientan hacia la programación personal o de oficina

> > > Spectrum, Atari 400 y 800, Vic-20, TI99/4A
> > BBC Modelo B, Commodore 64, Dragon-32, Sord M5, Tandy Color
> Oric-1



Atención, compradores
Todos los meses aumenta en las tiendas la oferta de nuevos ordenadores, hecho que dificulta aún más la decisión del recién iniciado acerca de qué modelo elegir. Con tantos factores a considerar, la vieja advertencia de "cuidado, comprador" es más apropiada que nunca

Tanto si está pensando en adquirir su primer microordenador personal como si ya posee uno y desea ahora mejorarlo para conferirle un mayor nivel de sofisticación, las opciones disponibles pueden dejarlo bastante desconcertado. En *Mi Computer* hasta ahora hemos descrito con todo detalle algunos de los ordenadores personales que existen en el mercado, hemos explicado los principios operativos y los métodos operativos básicos, hemos hablado acerca de algunas aplicaciones de los microordenadores en nuestra vida cotidiana y hemos definido algunos términos básicos del especializado lenguaje de la informática.

Al comprar por primera vez un microordenador se han de tener en cuenta muchos factores y, entre todos ellos, quizá el primero sea el relativo al precio de la máquina. Ahora uno puede acudir a unos grandes almacenes y adquirir un Sinclair ZX81 en su forma más sencilla por menos de 20 000 pesetas. El principal inconveniente del ZX81 es la calidad o el "tacto" de su teclado. Con el fin de reducir al máximo posible sus dimensiones físicas, Sinclair adoptó para el teclado un diseño de membrana de capas múltiples. El resultado es una cierta falta de sensibilidad o de "realimentación táctil", que se puede superar al entrar en contacto permanente con la máquina. El especial diseño de este microordenador lo dota de dos características: de un lado, gran simplicidad; de otro, cierta lentitud.

No obstante, existen otras características a tener en cuenta por el futuro comprador, como la imposibilidad del Sinclair ZX81 para generar colores o producir sonidos y su poca capacidad para admitir accesorios.

Estas mismas consideraciones son válidas para muchos otros ordenadores económicos: por ejemplo, el Jupiter Ace o el Sharp PC-1500 (aunque tal vez el Ace sea interesante si el comprador tiene previsto llegar a utilizar el FORTH), y por lo tanto no los hemos tenido en cuenta a la hora de confeccionar este informe.

Existe toda una serie de características que uno esperaría hallar en un micro personal. Éstas se han detallado en la página anterior y resulta vital reseñarlas en una lista personalizada, por orden de importancia. Si, como les ocurre a muchos usuarios, considera que lo utilizará prioritariamente para juegos por ordenador, entonces le interesará una amplia base de software, periféricos apropiados, como palancas de mando o mandos de bola, buenos gráficos y capacidad para generar sonido. Alternativamente, si su interés se centra en el campo de la administración empresarial-doméstica, entonces tal vez para usted revista más importancia el número de las columnas visualizadas en pantalla, la calidad del teclado, el almacenamiento de la información, una conexión sencilla para impresora y gran cantidad de software.

Si bien los ordenadores son muy versátiles, algunos se prestan mejor que otros para una tarea determinada. Es verdaderamente esencial que confeccione una lista exhaustiva antes de salir a adquirir su primera máquina. Incluya en esa relación todas las características que espera hallar en su ordenador; quizá también sea conveniente asignar a cada punto un valor de uno a diez por orden de prioridad.

Para una máquina de uso general busque una buena versión de BASIC, un teclado cómodo, claras posibilidades de ampliación y una memoria estándar razonablemente grande.

Vale la pena mirar antes un poco. Es poco probable que consiga que le presten mucha atención en una cadena de grandes almacenes, pero es posible que consiga convencer al vendedor de que lo deje probar uno o dos juegos. No olvide el mercado de segunda mano. Miles de personas han comprado ya ordenadores personales. Muchas de ellas desean mejorar su potencial

informático hasta un punto en que resulta mucho más lógico adquirir un ordenador nuevo. Si bien las operaciones de compraventa de máquinas electrónicas no son un hecho totalmente desconocido, todavía distan mucho de ser algo corriente. Eche una ojeada a las páginas de "Se vende" de las publicaciones de informática y a las de su periódico local.

Cuando comience a pensar en mejorar las prestaciones de su ordenador o en comprar periféricos, piense que es muy importante estudiar el mercado con mucha atención. Si tiene un Spectrum, por ejemplo, se verá limitado a los propios periféricos de Sinclair que, a pesar de ser bastante eficaces, tienen un margen limitado de posibilidades. O es posible que su intención sea aumentar la capacidad de memoria de su Commodore Vic-20, entonces se encontrará con que debe optar por alguna de las propias unidades de Commodore o decantarse por otras muy parecidas producidas por fabricantes independientes, que o bien ofrecen lo mismo por menos dinero o quizá una potencia ligeramente inferior por igual precio.

Tal vez el área que ofrece mayores posibilidades de elección sea la de las impresoras. No sólo existe una gran cantidad de fabricantes de estas máquinas, sino que también se encuentran alrededor de media docena de diferentes tipos, siendo las más comunes las matriciales, las de rueda margarita y las impresoras de chorro de tinta. Si usted centra su interés en el tratamiento de textos, probablemente le convendrá una impresora de tipo de diseño margarita, que produce unos resultados comparables a las copias obtenidas con las mejores máquinas de escribir eléctricas. No obstante, si tiene pensado realizar presupuestos por medio de un programa de hoja electrónica, el primer punto que habrá de tener en cuenta es la máxima longitud de línea.

En términos de amplitud de elección, el segundo lugar es el del almacenamiento de información. ¿Es conveniente invertir en una de las más caras unidades de disco flexible, o, por el contrario, preferir una cinta flexible (véase p. 224)? Una vez más le diremos que mientras más dinero destine a la compra de su ordenador y demás dispositivos electrónicos, mayores posibilidades habrá de que sus demandas sean satisfechas.



Primero, el software
A menos que piense utilizar su ordenador sólo para aprender a programar, la disponibilidad de un buen software aplicativo será tan importante como las especificaciones de aquél, si no más. Cuanto más se venda un ordenador, mayores son las posibilidades de que se desarrolle software para él, y cuanto más software haya, mejor se venderá la máquina: una verdadera situación del tipo de si fue antes el huevo o la gallina

Una de las cuestiones que más parecen preocupar al eventual comprador es: "¿Y si compro ahora y luego dentro de una semana el precio baja en un diez por ciento?" Todos los indicios hacen suponer que esta situación se prolongará en el futuro inmediato. Leyendo atinadamente las publicaciones de informática con frecuencia se pueden anticipar recortes de precio pero, en cualquier caso, ése no es un motivo válido para posponer la compra de un ordenador. Es mucho más importante que se esté seguro de que la máquina que se piensa adquirir es la adecuada, y, como hemos mencionado anteriormente, la forma más segura de acertar es pensar cuidadosamente en cuál será la aplicación principal a la que se la destinará. Recuerde siempre que su sistema informático es mucho más que la caja con teclado que se conecta con su aparato de televisión. Es más, incluso, que el software que haya adquirido y los periféricos. Su ordenador es una herramienta que, utilizada al máximo de su capacidad, puede pròporcionarle entretenimiento, resolverle problemas y llevarle registros concisos y de fácil acceso.

Ir de tiendas



Tiendas de descuento
Las tiendas que trabajan sobre la base de descuentos en equipos de alta fidelidad y videos suelen ofrecer microordenadores a buen precio

Grandes almacenes
Muchas personas compran sus ordenadores en los grandes almacenes, que los adquieren al por mayor. Al principio los vendedores de estos establecimientos tenían poco conocimiento acerca de estas máquinas; en la actualidad los grandes almacenes cuentan con departamentos de informática

Tienda especializada
Si le compra su ordenador a un comerciante profesional, puede estar seguro de contar con un eficiente servicio posventa y con consejos de absoluta confianza acerca del software, aunque es probable que no consiga el mejor precio

Venta por correo
Ésta solía ser una forma muy popular de vender microordenadores y, sin duda, el método más rentable para los fabricantes nuevos. Sin embargo, los largos períodos de espera para los nuevos ordenadores prometidos fueron la causa principal que determinó su pérdida de aceptación

Commodore Vic-20

La máquina más pequeña de la Commodore le ofrece al usuario unas configuraciones bastante sofisticadas a precio razonable

Commodore Business Machines (CBM) fue la responsable de uno de los primeros micros personales (el Personal Electronic Transactor: PET), que salió al mercado en 1977. En 1981 lanzó el Commodore Vic-20, que incorpora muchas características del PET. El Vic no sólo utiliza el mismo microprocesador 6502, sino incluso el mismo BASIC en ROM que, lamentablemente, presenta el inconveniente de que no se trata de la versión más eficaz ni la más reciente de las producidas por Commodore.

La diferencia más evidente entre las dos máquinas concierne a las capacidades para gráficos adicionales del Vic. Su nombre proviene del chip exclusivo que activa su visualización en pantalla: el Video Interface Chip. Ofrece una disponibilidad de hasta 16 colores, aunque la visualización consta de un marco o reborde para el cual hay ocho colores disponibles; un fondo, que puede ser de cualquiera de los 16 colores; y los caracteres o símbolos individuales, cuyos colores se pueden escoger entre un grupo de ocho.

El juego de caracteres en sí mismo es asombrosamente grande, ofreciendo tanto mayúsculas como minúsculas y dos juegos de caracteres para gráficos a partir de 62 teclas, además de cuatro teclas exclusivas que se pueden utilizar, pulsando SHIFT, para proporcionar ocho funciones programables. El diseño del teclado es particularmente acertado, tanto ergonómica como técnicamente.

El principal inconveniente del Vic es su pequeña capacidad de memoria: sólo cinco Kbytes, que se reducen a 3,5 Kbytes después de que el sistema operativo ha utilizado RAM para la pantalla y para otras exigencias internas. Sin embargo, tiene la ventaja de que es posible direccionar hasta 32 Kbytes de memoria, además de lo cual se puede disponer de memoria adicional desde diversas fuentes.

Se proporcionan conexiones para interfaces para mandos de raqueta, palanca de mando, lápiz óptico, cartuchos de juegos, ampliación de memoria, impresora, unidad de disco, cassette y televisor, y hay una que responde al estándar en serie RS232, que se puede emplear con un modem o con una impresora que no sea Commodore. Además de todo ello, existe una amplia gama de accesorios de hardware, los mismos que para el más reciente Commodore 64 (véase la sección "Modelos de Hardware" de página 49).

Conexión cassette
El Vic-20, al igual que otros micros Commodore, necesita de una grabadora de cassette fabricada especialmente, que se enchufa aquí

Conexión dispositivos
Este conector de 24 patillas es en serie y se utiliza para activar ciertos dispositivos periféricos adicionales

Adaptador para interface de periféricos
Estos chips controlan todas las operaciones de input/output del Vic-20, y poseen capacidad de procesamiento propia. Pueden, por ejemplo, efectuar la conversión entre formatos en serie y en paralelo

Conexión teclado
Aquí se conecta el teclado con el adaptador para interface de periféricos

Chip para interface de video
Este dispositivo incorporado especial se utiliza para controlar el estado de la pantalla y también el generador de sonido de tres voces del Vic-20

Memoria de acceso directo (RAM)
El Vic-20 está dotado de una RAM de 5 K, la cual se puede ampliar externamente a 32 K



Los periféricos del Vic
Commodore suministra una unidad de cassette, una unidad de disco y una gama de impresoras para el Vic-20, así como palancas de mando. Alternativamente, hay una gran variedad de dispositivos de otros fabricantes. De éstos, tal vez los más importantes sean los módulos para ampliación de memoria



Teclado
Las 62 teclas del teclado proporcionan cuatro juegos completos de caracteres, y permiten la utilización de hasta ocho funciones programables en virtud de las teclas exclusivas situadas a la derecha

Chris Stevens

Carga de cartuchos
Esta ranura acepta cartuchos de software ROM o placas de ampliación de memoria

Enchufe red
Aquí se conecta la unidad de alimentación eléctrica externa

Interruptor

Conexión juegos
Aquí se pueden conectar lápices ópticos, palancas de mando o mandos de raqueta

Enchufe audio-video
Aquí se conecta un modulador de RF externo

Conexión en serie
La mayoría de los periféricos del Vic, la impresora y la unidad de disco, por ejemplo, se accionan desde esta conexión diseñada especialmente que, a diferencia de la mayoría de las interfaces en serie, puede direccionar más de un dispositivo periférico a la vez

Microprocesador
El Vic-20 utiliza el microprocesador 6502 de MOS Technology, de probada eficacia y enorme popularidad

Memoria de lectura solamente (ROM)
Estos chips contienen el BASIC, el juego de caracteres y toda la información necesaria para interpretar y ejecutar los programas

Commodore Vic-20

DIMENSIONES
404 × 216 × 75 mm

PESO
1 820 g

VELOCIDAD DEL RELOJ
1 MHz

MEMORIA
La máquina viene con 5 K de memoria estándar ampliables a 32 K por medio de un adaptador de 16 K, otro de 8 K y un tercero de 3 K

VISUALIZACION EN VIDEO
23 líneas de 22 caracteres. Los gráficos, de alta resolución, permiten 184 × 175 pixels. Ofrece un máximo de 16 colores

INTERFACES
Dos conexiones en serie, enchufe audio-video, conexión cassette, carga de cartuchos, conexión juegos

LENGUAJE SUMINISTRADO
BASIC

OTROS LENGUAJES DISPONIBLES
Ensamblador y órdenes adicionales en BASIC

VIENE CON
Unidad de alimentación eléctrica, cable para antena, manual

TECLADO
Teclado de tamaño normal similar al de una máquina de escribir, con 62 teclas y otras cuatro de función especial

DOCUMENTACION
Una de las áreas menos cuidadas de la CBM es la de la documentación que ofrece con sus ordenadores personales. Los sistemas de gestión como el 8032 vienen con un ejemplar de la excelente obra de Adam Osborne "Guide to CBM Computing", pero a los propietarios del Vic se los relega a un plano mucho más trivial. Aunque está escrito con sencillez y, por consiguiente, resulta fácil de comprender, el manual del usuario apenas si se queda en la superficie de las configuraciones de la máquina. Afortunadamente, se encuentran a la venta gran cantidad de publicaciones que pueden cumplir un papel alternativo

Consultando la agenda

Valiéndonos de todas las técnicas de la programación en BASIC que hemos aprendido hasta el momento, damos ahora los primeros pasos para desarrollar un programa de base de datos

Ahora que ya hemos cubierto muchos de los fundamentos del BASIC, es el momento de poner en práctica cuanto hemos aprendido desarrollando un verdadero programa. Por supuesto, todos los programas con los cuales nos habíamos encontrado hasta ahora eran también programas "verdaderos" en el sentido de que realizaban una tarea específica, pero eran ilustraciones de cómo funcionan los diversos componentes del lenguaje BASIC y no la clase de programas que el usuario desearía emplear todos los días. Ilustraban cómo los "engranajes" del BASIC se podían unir entre sí para formar un mecanismo sencillo. En este momento estamos en disposición de ensamblar esos mecanismos. ¡Ahora es ya un reloj completo lo que vamos a construir!

Una de las preguntas más comunes que las personas que no emplean ordenadores les formulan a los usuarios de ordenadores es: "¿Para qué se puede utilizar realmente un ordenador?" La pregunta no es tan simplista como parecería a primera vista. Las respuestas más convencionales tienden a seguir la línea de: "Bueno, puedes informatizar tus recetas de cocina" o "Puedes crear una agenda de direcciones o de teléfonos informatizada". Es muy raro que esta táctica dé buenos resultados, porque quien hizo la pregunta por lo general hace comentarios como: "Pero yo puedo consultar mi libro de cocina cada vez que deseo cocinar algo y puedo leer mi agenda de direcciones cuando quiero hallar la dirección de alguien, sin necesidad de tener que pasarme horas escribiendo un programa para hacerlo". ¿Qué circunstancias se deben considerar para decidir si a un problema se le debe buscar respuesta por medio de la informática y no por métodos convencionales? Responderemos a esta pregunta trabajando en el ejemplo específico de una agenda de direcciones informatizada.

Una agenda de direcciones corriente comúnmente consta de un índice alfabético que se puede manipular con los dedos, diseñado para que el usuario pueda localizar, muy aproximadamente, un nombre determinado. Por lo general, y cuando es necesario, se suelen ir agregando otros nombres, y no por orden estrictamente alfabético. Su primera entrada en la página de la P podría ser Daniel Puig. Después se podría agregar Alfonso Prado o Camilo Pérez. Aunque estos nombres no está ordenados alfabéticamente, están agrupados juntos bajo la P, de modo que la tarea de hallar un apellido determinado que comience con P no será demasiado complicada. Por otra parte, si no se utilizara ningún tipo de índice, localizar un nombre sería un verdadero problema.

Las otras entradas corrientes en una agenda de direcciones son las señas y el número de teléfono de la persona e, incluso, alguna información personal. Sin embargo, una agenda de direcciones convencional no le puede dar una lista separada de todas las personas que viven en Barcelona, o a quién le corresponde un determinado número de teléfono. Puede que esto no represente un inconveniente grave, pero si usted fuera propietario de una pequeña empresa de ventas por correo sería una ayuda muy valiosa que pudiera obtener información específica acerca de las personas de su cartera de clientes. Por ejemplo, si quisiera distribuir una nueva línea de pijamas para niños podría anticipar nuevos pedidos informando a sus clientes, pero para ahorrarse algún dinero en sellos de Correo probablemente no valdría la pena enviarles los folletos a los clientes que no tuvieran hijos. Éste es el tipo de consideraciones que se deben evaluar antes de decidir si, frente a un problema dado, es más conveniente una solución informática o una solución convencional.

Si la solución informática es apropiada, la siguiente consideración ha de ser, entonces, si comprar o no un software existente a nivel comercial. Una mirada a los anuncios de las revistas de informática sugeriría que los programadores de ordenadores ya han pensado en todas las eventualidades posibles. Sin embargo, un examen más detenido podría dejar en evidencia el hecho de que un programa disponible comercialmente podría no ajustarse con toda exactitud a lo que se desea, o que no exista para su modelo de ordenador, o que sea demasiado caro. El costo de un programa refleja generalmente los costos de desarrollo. Un paquete de tratamiento de textos puede ser caro, pero si el usuario decidiera escribirlo él mismo y tuviera que destinar seis meses de dedicación exclusiva a su realización, sin duda le resultaría mucho más caro.

Desde el punto de vista de lo positivo, el software que se escriba usted mismo podría hacer exactamente lo que deseara que hiciera. El otro factor es la incomparable satisfacción de que, solo y sin ayuda de nadie, escriba usted un programa con el que obtenga un éxito rotundo.

El diseño de un programa consta de varias fases, la primera de las cuales es la comprensión profunda del problema, premisa que implica una descripción clara de dicho problema.

La segunda consiste en hallar un enfoque para solucionar el problema. Esto implica una descripción de la forma esperada de la entrada y la salida como un "primer nivel de descripción" del problema. Los problemas y las soluciones se deben plantear en los términos más amplios y éstos se han de concretar gradualmente hasta llegar a la etapa en la cual los podemos codificar en un lenguaje determinado.

La tercera etapa es la codificación en sí misma. Utilizaremos el BASIC como nuestro lenguaje de alto nivel, pero en lugar de éste podríamos emplear igualmente cualquier otro lenguaje. Hasta la etapa final de codificación en BASIC utilizaremos un seudolenguaje intermedio entre la libertad y flexibilidad del lenguaje corriente y las estructuras rígidas de un verdadero lenguaje para ordenador como el BASIC.

El enfoque a la programación que acabamos de des-

cribir se denomina generalmente programación *top-down* (de arriba/abajo). Se trabaja desde el nivel superior (una enunciación general de los objetivos globales), a través de varios niveles de refinamiento, hacia abajo, hasta los detalles más precisos del programa necesarios para empezar a codificar en el lenguaje de alto nivel escogido. También intentaremos ceñirnos a los principios de la denominada "programación estructurada". Estos principios se irán clarificando a través del desarrollo de este proyecto.

Los pasos que seguiremos en el desarrollo del programa se pueden resumir de la siguiente manera:

1. Una clara enunciación del problema
2. La forma de la entrada y la salida (primer nivel de descripción)
   - 2.1 Refinamiento (segundo nivel de descripción)
   - 2.2 Sucesivo refinamiento (del tercer al enésimo nivel de descripción)
3. Codificación en lenguaje de alto nivel

Antes de comprometernos en un proyecto importante de software, es esencial plantear el problema con toda claridad. Se trata de un ejercicio que no es en absoluto trivial. Probemos con algunas ideas para nuestra agenda de direcciones informatizada.

Empezaremos primero por una lista de las características deseables; luego podemos decidir cuáles de ellas se pueden emplear con un esfuerzo de programación razonable. Deseamos que se pueda:

1. Localizar una dirección, un número de teléfono y notas dando entrada a un nombre desde el teclado
2. Obtener una lista de nombres, direcciones y números de teléfono dando entrada a sólo parte de un nombre (sólo el primer apellido o el nombre)
3. Obtener una lista de nombres, direcciones y números de teléfono de una ciudad o unas zonas determinadas
4. Obtener un listado de todos los nombres que comienzan por una letra determinada
5. Obtener un listado completo de todos los nombres de la agenda de direcciones, ordenados alfabéticamente
6. Agregar todas las entradas nuevas que deseemos
7. Modificar la información ya existente
8. Suprimir los datos que deseemos

Supongamos que ya hemos escrito el programa para la agenda de direcciones. ¿Qué forma deberían asumir la entrada y la salida? ¿Cómo le gustaría que funcionara el programa desde el punto de vista del usuario? En términos generales, los programas pueden ser "activados por menú", "activados por órdenes" o por una combinación de ambos. En un programa activado por menú, cada vez que se ha de tomar una decisión se le ofrece al usuario una lista (menú) de opciones. La selección se suele efectuar pulsando sólo una tecla. En los programas activados por órdenes, el usuario digita las palabras-órdenes o frases-órdenes específicas, por lo general sin que se le requiera en ese sentido. Algunos programas combinan ambas técnicas. La ventaja de un programa activado por menú es que a un recién iniciado le debería resultar sencillo de utilizar, logrando que el programa fuera más "amable con el usuario". Un programa activado por órdenes debería resultarle más expedito al usuario experimentado. Nosotros optaremos por un enfoque activado por menú, si bien usted está en libertad de decidirse por ejecutar este programa utilizando, en cambio, rutinas de órdenes: la decisión es suya.

Dado que el programa se centrará en una lista de nombres, lo primero que debemos considerar es qué forma deben asumir esos nombres. ¿Comprenderá el ordenador, por ejemplo, todos los formatos siguientes?:

A. J. P. Soler
Leonardo da Vinci
Bo DEREK
Juana R.
L. Prado
j. j. García
F López
Twiggy
GROUCHO MARX
Sir Freddie Smith

Esto puede sonar como que estamos hilando muy fino, pero consideremos lo que sucedería si diera entrada a `F López` y después le pidiera al programa que buscara `F. López`. A menos que usted hubiera previsto el problema, probablemente el ordenador le respondería `NOMBRE NO HALLADO`.

Existen dos formas de afrontar el problema: podemos tener una entrada "libre", que permita dar entrada a los nombres en cualquier forma junto con rutinas inteligentes que tomen en consideración este hecho cuando se realiza la búsqueda; o bien podemos insistir en que se dé entrada a los nombres en una forma estrictamente definida. Todo nombre que no se ciñera a ésta produciría un mensaje de error como `FORMATO DE NOMBRE INACEPTABLE`. La elección es de índole arbitraria, pero tenemos la posibilidad de inclinarnos por la opción de una entrada muy "libre" y dejar que el programa se preocupe de convertir los nombres en una forma estándar.

Desde el punto de vista de una búsqueda alfabética, podemos imaginar que los nombres poseen dos partes: el nombre de pila y el primer apellido. Un apellido es relativamente sencillo de definir: una serie de caracteres alfabéticos en mayúsculas o minúsculas terminados en un Carriage Return (retorno de carro) y precedidos por un espacio (ASCII 32). De inmediato se presenta un problema: ¿qué sucedería si se diera entrada sólo al nombre "Twggy" sin estar precedido por un espacio? Presumiblemente el programa lo rechazaría por tener un formato inaceptable. Será mejor que modifiquemos nuestra definición.

Consideremos que un nombre consta de una o dos partes: el apellido o el apellido y el nombre de pila. El nombre puede incluir tanto caracteres en mayúsculas o en minúsculas, como puntos, apóstrofos y guiones. Siempre comenzará con un carácter alfabético y terminará con un "Carriage Return" (no se admitirán puntos al terminar). De haber un espacio, el último grupo de caracteres (incluyendo apóstrofos y guiones) se contará como el apellido y otras partes, incluyendo el espacio, se contarán como el nombre de pila. De no haber espacio, el nombre completo se considerará como el apellido.

El apellido requiere una consideración especial porque, en el ordenamiento alfabético, precede siempre a los nombres de pila. De manera que Ana Pérez vendría después de Victoria Paredes. Si un nombre se compone de solo un grupo de caracteres, como Trevanian, Twiggy o un apodo como "Gordo", a los efectos de nuestro programa se lo puede considerar como un apellido.

En una búsqueda alfabética, ¿qué nombre iría primero: A.J.P. Soler o Alfredo Soler? La decisión es arbitraria, pero la solución más sencilla sería ignorar los caracteres no alfabéticos comprendidos antes del último espacio y hacer que los nombres equivalgan a AJP Soler y ALFREDO SOLER. Al proceder de esta manera, tanto ALFREDO como AJP se considerarían como nombres de pila, y AJP iría primero.

Parte de nuestro programa aceptaría como entrada un nombre y produciría como salida un nombre, una dirección y un número de teléfono (observe que aún no hemos ni siquiera comenzado a considerar los significados de "dirección" y "número de teléfono"). Si aceptáramos como entrada nombres de formato "libre", con conversión interna a un formato estandarizado, ¿esperaríamos que la salida estuviera en la forma "estandarizada" o en la misma forma que la entrada original? La salida más cómoda para el usuario sería aquella en la que el nombre estuviera en la forma original, pero, como veremos, esto complicará la programación.

Como tarea inicial de programación, supongamos que se le ha asignado un nombre a la variable alfanumérica NOMBRE$ (nombre completo) y que tenemos otras dos variables, NOMB$ (nombre de pila) y APELL$ (apellido). ¿Cómo le asignaremos a NOMB$ y a APELL$ las partes apropiadas de NOMBRES? Ignorando, de momento, el problema de llevar un registro de la forma original en que se ha dado entrada al nombre (de modo que se pueda recuperar cuando la necesitemos más adelante), una sentencia simple del programa podría ser:

Convertir en mayúsculas todos los caracteres
Eliminar todos los caracteres no alfabéticos excepto el espacio final
Asignar a APELL$ todos los caracteres que siguen al último espacio
Asignar a NOMB$ todos los caracteres que preceden al último espacio

Antes de considerar cómo se podría codificar este problema en BASIC, veremos cómo el proceso de "programación de arriba abajo" nos puede llevar desde una enunciación muy amplia de nuestro objetivo hasta el punto en el que se hace posible la codificación en un lenguaje de programación determinado. Observará que estamos utilizando no sólo nombres de variables largos como NOMBRES, sino palabras-órdenes como BEGIN, LOOP y ENDLOOP. Éstas son construcciones que hemos inventado para que nos ayuden a describir nuestro programa. En la etapa final de desarrollo serán reemplazadas por órdenes equivalentes en BASIC. Explicaremos mejor estas órdenes, así como por qué hemos sangrado algunas de las líneas, en el próximo capítulo de nuestra obra.

1.ª ENUNCIACION DE OBJETIVOS

```
INPUT
Un nombre (en cualquier formato)
OUTPUT
1. El nombre de pila
2. El apellido
```

1.er REFINAMIENTO

```
1. Leer NOMBRE$
2. Convertir en mayúsculas todas las letras
3. Hallar último espacio
4. Leer APELL$
5. Leer NOMB$
6. Eliminar caracteres no alfabéticos de NOMB$
```

2.º REFINAMIENTO

```
1. Leer NOMBRE$
2. (Convertir en mayúsculas todas las letras)
   BEGIN
   LOOP mientras los caracteres no explorados
   permanecen en NOMBRE$
     Leer los caracteres de NOMBRE$ sucesivamente
     IF el carácter está en minúscula
       THEN convertir en mayúscula
       ELSE no hacer nada
     ENDIF
     Asignar el carácter a la variable alfanumérica
     transitoria
   ENDLOOP
   LET NOMBRE$ = variable alfanumérica transitoria
   END
3. (Hallar último espacio)
   BEGIN
   LOOP mientras los caracteres no explorados
   permanecen en NOMBRE$
     IF carácter = " "
       THEN anotar posición en una variable
       ELSE no hacer nada
     ENDIF
   ENDLOOP
   END
4. (Leer APELL$)
   BEGIN
   Asignar a APELL$ los caracteres a la derecha del último
   espacio de NOMBRE$
   END
5. (Leer NOMB$)
   BEGIN
   LOOP mientras caracteres no explorados permanecen en
   NOMBRE$ hasta último espacio
     SCAN caracteres
     IF el carácter no es una letra del alfabeto
       THEN no hacer nada
       ELSE asignar el carácter a NOMB$
     ENDIF
   ENDLOOP
   END
6. (Eliminar caracteres no alfabéticos de NOMB$)
   (Esto se ha manipulado arriba, en 5)
```

Este segundo nivel de refinamiento está ahora muy cerca de la etapa donde será codificado en un lenguaje de programación. Desarrollemos ahora 2 (es decir, convertir en mayúsculas todas las letras) en un tercer nivel de refinamiento y codifiquémoslo a continuación en lenguaje BASIC. Anteriormente ya habíamos encontrado un algoritmo que nos permitía realizar este proceso (véase p. 212).

3.er REFINAMIENTO

```
2. (Convertir en mayúsculas todas las letras)
   BEGIN
   READ NOMBRE$
   LOOP
   FOR L = 1 TO longitud de la variable
     READ carácter L
     IF carácter está en minúscula
       THEN restarle 32 al valor del
         carácter en ASCII
       ELSE no hacer nada
     ENDIF
     LET VARTRAN$ = VARTRAN$ + carácter
   ENDLOOP
   LET NOMBRE$ = VARTRAN$
   END
```

Este fragmento de programa en seudolenguaje ahora ya se parece bastante a un lenguaje de programación como para ser codificado. Nuestra versión de BASIC Microsoft no permite que los nombres de variables alfanuméricas sean palabras completas, de modo que en su lugar usaremos letras. Por lo tanto, NOMBRE$ se convierte en N$.

```
1000 REM SUBRUTINA DE CONVERSION A MAYUSCULAS
1010 INPUT "DE ENTRADA A NOMBRE";N$: REM SOLO
     PARA PROBAR
1020 LET P$ = " ": REM ASEGURAR QUE LA VARIABLE
     ESTA VACIA
1030 FOR L = 1 TO LEN(N$): REM INDICE DE BUCLE
1040 LET T$ = MID$(N$,L,1): REM EXTRAER
     CARACTER
1050 LET T = ASC(T$): REM HALLAR VALOR ASCII DE
     CARACTER
1060 IF T > = 97 THEN LET T = T - 32: REM
     CONVERSION A MAYUSCULAS
1070 LET T$ = CHR$(T)
1080 LET P$ = P$ + T$: REM P$ ES LA VARIABLE
     TRANSITORIA
1090 NEXT L: REM FIN BUCLE
1100 LET N$ = P$: REM AHORA N$ ESTA TODA EN
     MAYUSCULAS
2000 PRINT N$: REM SOLO PARA PROBAR
2010 END: REM SOLO PARA PROBAR. REEMPLAZAR
2020 REM 'RETURN' EN PROGRAMA VERDADERO
```

Este fragmento de programa se podría usar como una subrutina dentro de un programa principal. Lo hemos escrito para probar con una sentencia INPUT, otra PRINT, y varias REM y un END. No obstante, estas dos últimas habrían de eliminarse antes de que la subrutina se incorporara a un programa completo.

Complementos al BASIC



El Lynx posee una función, UPC$(A$), que convierte en mayúsculas las letras de A$, de manera que el programa se reduce así:

```
1010 INPUT "dé entrada a un nombre";N$
1020 LET N$ = UPC$(N$)
2000 PRINT N$
2010 REM "RETURN" AQUI EN EL
     PROGRAMA VERDADERO
```



El programa completo para el Spectrum es:

```
1010 INPUT "DE ENTRADA A UN
     NOMBRE";N$
1020 LET P$ = " "
1030 FOR L = 1 TO LEN N$
1040 LET T$ = N$(L)
1050 LET T = CODE T$
1060 IF T > = 97 THEN LET T =
     = T - 32
1070 LET T$ = CHR$ T
1080 LET P$ = P$ + T$
1090 NEXT L
2000 LET N$ = P$
2010 PRINT N$
2020 REM "RETURN" AQUI EN EL
     PROGRAMA VERDADERO
```

DRAGON 32

Este programa se puede ejecutar en el Dragon, pero los caracteres en minúsculas están reservados para la impresora, de modo que el problema no se plantea realmente



Reemplazar la línea 1010 por:

```
1010 INPUT "dé entrada a un nombre",
     N$
```

Ejercicios

■ Refinar todas las etapas anteriores hasta el punto en que se pudieran convertir en un programa en BASIC. El "seudolenguaje" que emplee no tiene por qué ser igual al nuestro, pero es una medida atinada continuar utilizando letras mayúsculas para los términos clave que probablemente corresponderán a palabras-sentencia en el lenguaje final (por ejemplo, LOOP, IF, LET, etc.). Emplee letras pequeñas para las operaciones que habrán de enunciarse más explícitamente cuando se codifiquen finalmente. Éstas las puede escribir en lenguaje corriente.

■ Luego de haber desarrollado los programas hasta un nivel de refinamiento satisfactorio, conviértalos en módulos de programa (subrutinas) en BASIC. Verifíquelos uno a uno empleando entradas ficticias e imprima sentencias que, si funcionan adecuadamente, se puedan eliminar luego.

Ejercicios de revisión

En estos ejercicios se aplican la mayor parte de las sentencias y funciones en BASIC que se utilizan más comúnmente. No hay ninguna pregunta capciosa ni se introduce ningún concepto nuevo. Si puede realizar todos estos ejercicios o la mayoría de ellos sin ninguna dificultad, puede usted considerar que está ya en camino de convertirse en un programador en BASIC hecho y derecho.

■ Escriba un programa para aceptar una entrada de dos números desde el teclado, para sumarlos e imprimir el resultado.

■ Asigne dos palabras (series de caracteres) a dos variables alfanuméricas y luego cree una tercera variable alfanumérica que concatene (es decir, que una entre sí) las dos palabras originales. Imprima a continuación la tercera variable.

■ Escriba un programa que le permita digitar cualquier palabra en el teclado y que después imprima la longitud de la variable en el mensaje LA PALABRA QUE HA DIGITADO TIENE * CARACTERES (* representa un número).

■ Escriba un programa que acepte un único carácter digitado en el teclado y que le diga luego cuál es el valor ASCII de ese carácter (en decimal).

■ Escriba un programa que le ofrezca el mensaje DIGITE UNA PALABRA y que después le responda con el mensaje LA ULTIMA LETRA DE LA PALABRA ERA * (* representa una letra).

■ Escriba un programa que le solicite que DIGITE EL NOMBRE Y PRIMER APELLIDO DE UNA PERSONA y que después le responda EL ESPACIO ERA EL *º CARACTER.

■ ¿Cómo modificaría el programa anterior para que imprimiera 3^{er} en vez de 3^{o} si el espacio estuviera en la tercera posición?

■ Escriba un programa que le solicite que DIGITE UNA ORACION y que después le responda con el mensaje LA ORACION QUE DIGITO TENIA * PALABRAS (dando por sentado que en la oración habrá una palabra más que el número de espacios).

■ Controle su ordenador para ver si se le han asignado caracteres (o gráficos especiales) a los valores de ASCII desde 128 hasta 255 (utilice un bucle y la función CHR$(X)).

La sala de espera

Los ordenadores transfieren la información a mayor velocidad que la que pueden manipular los dispositivos mecánicos. Ello se obvia con una memoria intermedia, el "buffer" o tampón

Los *buffers* (topes) que utilizan los trenes están diseñados para amortiguar el impacto, absorbiendo energía en muelles o pistones amortiguadores. Los ordenadores también poseen buffers y de alguna forma funcionan como aquellos topes, pues ayudan a "permanecer juntos" a los componentes del sistema del ordenador.

En el mundo de la informática este término se aplica con cierta vaguedad y se lo emplea en dos sentidos bastante diferentes. Para el programador, buffer significa una utilización especializada de memoria del ordenador, mientras que para el diseñador de circuitos significa una clase de amplificador de señales eléctricas. Los del segundo tipo, que denominaremos *buffers de señal*, son los de las tablas de conexiones.

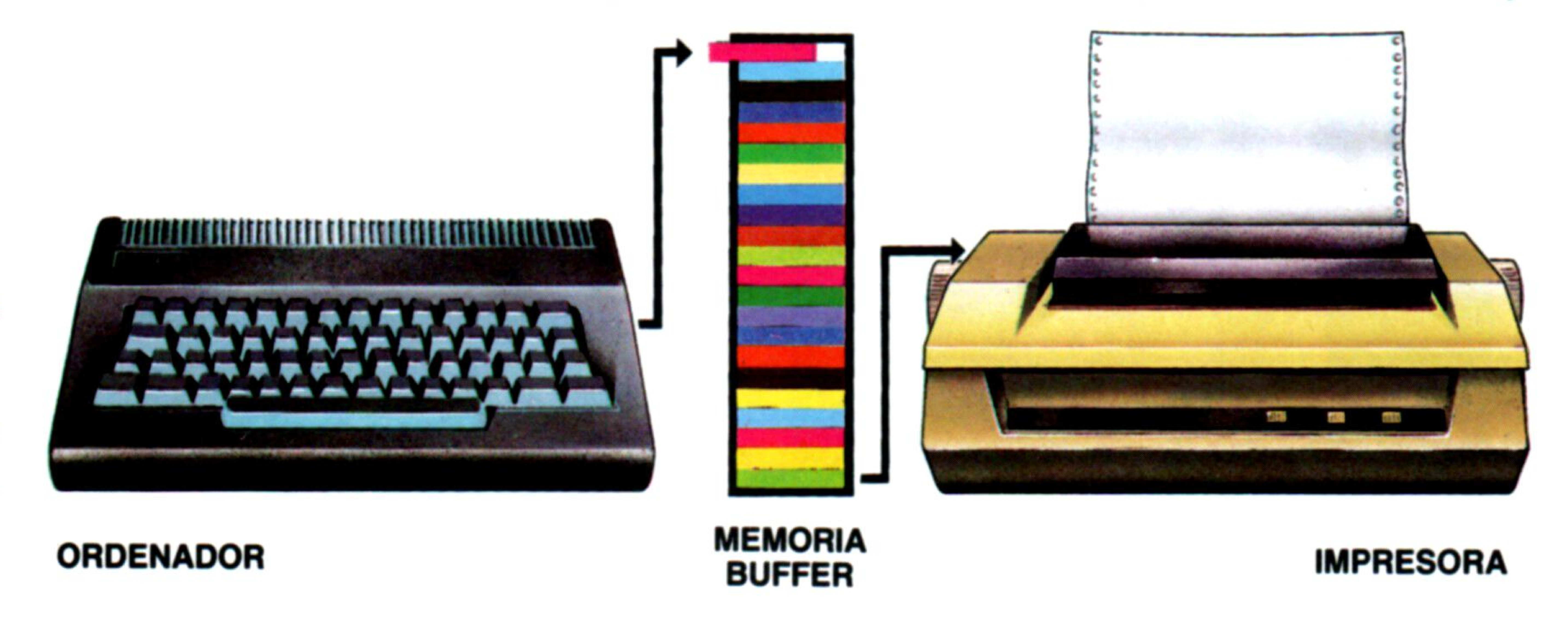
Buffers de memoria

Pensemos en un programa para tratamiento de textos que, entre otras cosas, puede desplazar un bloque de texto desde una parte de un "documento" en la memoria del ordenador hasta otra. El texto se compone de caracteres imprimibles y espacios, y de ciertos caracteres "no imprimibles", como el Carriage Return (retorno de carro). En la memoria del ordenador todos ellos están representados como códigos ASCII en binario. Para cada carácter se requiere un byte de memoria. Llevar los caracteres del bloque desde su antigua posición en la memoria hasta una nueva implica que se debe apartar otra parte de la memoria del ordenador para que sirva de zona de almacenamiento temporal de texto. Dicha zona de memoria apartada para una tarea específica se denomina *buffer*, o tampón.

Como segundo ejemplo, consideremos el problema de imprimir un documento creado con un procesador de textos. El documento podría constar de 15 000 caracteres separados, pero es evidente que no se le podrían enviar todos simultáneamente a la impresora para que los imprimiera: la velocidad de impresión de la mayoría de estos dispositivos no suele superar la cifra aproximada de 80 caracteres por segundo. Para orillar este inconveniente, se apartará parte de la memoria del ordenador, que cumplirá la función de *buffer de impresión*, bajo el control del software de tratamiento de textos. Este programa llenará primero este buffer con los caracteres a imprimir, y luego los irá enviando para su impresión a una velocidad adecuada para la impresora.

El buffer de impresión puede no ser muy grande, tal vez de una capacidad de sólo 128 a 256 bytes, pero, independientemente de sus dimensiones, los principios que lo rigen son los mismos. Primero se escribe en él un "bloque" de caracteres ASCII y después éstos se vuelven a enviar de uno en uno. El primer byte que se escriba en el buffer será, asimismo, el primer byte que se lea de él (como es lógico, deseamos que los caracteres se impriman en el mismo orden en que fueron digitados). Esta clase de buffer se conoce como buffer FIFO (*First In First Out*: primero en entrar, primero en salir). Una vez leídos todos los caracteres del buffer, el software lo llena con el siguiente bloque de caracteres destinados a la impresora.

Los buffers FIFO son muy empleados en la mayor parte de este tipo de software. Se utilizan cuando existe incompatibilidad de velocidades, no sólo entre ordenadores e impresoras, sino también entre ordenadores y unidades de disco flexible y entre ordenadores y teclados de ordenador. Aunque la asombrosa velocidad de procesamiento de los ordenadores significa que por lo general éstos pueden identificar las teclas digitadas en un tiempo menor que el empleado en pulsarlas, es posible que en algunas ocasiones el ordenador no pueda identificar las teclas y visualizar los caracteres correspondientes con la suficiente rapidez. Esto puede suceder si el ordenador está momentáneamente ocupado cumpliendo otra función (accediendo a un disco, por ejemplo). Cuando esto ocurre, lo común es incorporar en el sistema operativo del ordenador un *buffer de digitación adelantada*. Este buffer "recuerda" cuáles teclas se han pulsado y el ordenador las visualiza de inmediato. Lo más probable es que esta acción pase inadvertida al usuario, pero con ciertos sistemas ope-



Parada temporal
Uno de los usos más comunes del buffer es aquél entre el ordenador y una impresora, dado que ésta no puede dar salida a los caracteres a la misma velocidad en que el ordenador los envía. En consecuencia, los caracteres se almacenan en la memoria temporal hasta que este buffer esté completo, y entonces se le envía al ordenador una señal de "ocupado" para que éste deje de transmitir. Los contenidos de la memoria buffer se envían luego a la impresora por el mismo orden en que se recibieron, pero a una velocidad muy inferior. Cuando esta tarea ha terminado, el proceso vuelve a comenzar hasta que se haya impreso el texto completo



Si echamos agua con un cubo dentro de un tanque, del grifo de salida saldrá un chorro uniforme. Del mismo modo, el ordenador "vierte" datos en el buffer, el cual los libera a una velocidad muy inferior a la que funciona, por ejemplo, la impresora

rativos de disco, o con ciertos tipos de software aplicativo (que implican mucho procesamiento de información), podría producirse un ligero intervalo entre la pulsación de la tecla y su aparición en pantalla. Unos pocos sistemas operativos permiten encender o apagar el buffer de digitación adelantada, o incluso que el usuario altere las dimensiones del buffer.

La forma exacta en que está organizado el software para manipular los buffers varía según la función que hayan de cumplir los mismos, pero por lo general será necesario apartar unos pocos bytes para utilizarlos como contadores y banderas. Es indispensable saber cuántos bytes se han leído de un buffer lleno antes de escribir otros; de lo contrario, se podrían destruir datos importantes antes de que se los utilizara.

Buffers de memoria en hardware

Los lectores que posean impresoras pueden haber notado lo lentas que éstas parecen ser, especialmente al imprimir un listado de programa o un documento largo. La mayoría de los sistemas operativos de ordenador no pueden realizar ninguna otra labor mientras se está utilizando la impresora; por lo tanto, si la impresión requiere mucho tiempo, el usuario no tendrá otra opción que permanecer sentado frente a la pantalla a la espera de que la impresora finalice su cometido. Ahora muchos fabricantes ofrecen buffers de impresión accesorios, por lo general en forma de una caja que se conecta entre el ordenador y la impresora. En efecto, desde el punto de vista del ordenador, estas cajas consiguen que la impresora trabaje más aceleradamente. En realidad ésta no imprime con mayor rapidez, sino que esta mayor "velocidad" se debe a la memoria extra exclusiva (a veces hasta de 16 Kbytes), con su propio software incorporado, que poseen estas cajas. Cuando el ordenador ha de imprimir un archivo, le traspasa bytes a la impresora hasta que recibe una señal de "ocupado", que significa que ésta ya no puede aceptar ningún byte más. El ordenador, entonces, ha de esperar hasta que la línea "ocupada" se convierta en "falsa", lo que indicará que la impresora está nuevamente en condiciones de aceptar datos. Si bien por lo general las impresoras cuentan con un pequeño buffer de memoria incorporado, éste no suele poseer más de dos Kbytes y no permite que el ordenador envíe más datos hasta encontrarse vacío. Los accesorios de buffers de memoria en hardware contienen más memoria, de modo que pueden aceptar muchos más datos antes de enviarle al ordenador la señal de "ocupado". Si el buffer es lo suficientemente grande, puede tener capacidad para retener de una sola vez todos los datos a imprimir, con lo cual el ordenador podrá continuar con otras tareas mientras el buffer le envía los datos a la impresora a menor velocidad.

Con frecuencia la memoria se utiliza más bien como un gran soporte de archivo o bus para almacenar programas y datos, pero en lugar de ello se puede organizar en "pilas" o buffers. Las "pilas" son estructuras LIFO (*Last In First Out*: último en entrar, primero en salir), mientras que los buffers, como acabamos de ver, son estructuras FIFO (*First In First Out*: primero en entrar, primero en salir). Para las pilas se suele emplear la analogía del montón de platos apoyados sobre un resorte que existe a menudo en los mostradores de los self-services. Los platos están apilados en el rimero y el último que se coloque en él será el primero que se quite. Al igual que los buffers, las "pilas" también son zonas de memoria temporal y sólo se diferencian de los buffers en el orden en que se da entrada y se recupera la información. En los lenguajes de alto nivel (como en los intérpretes de BASIC, por ejemplo), las "pilas" se utilizan "internamente", pues se necesita almacenar temporalmente la información para ser cargada luego. Consideremos este fragmento de programa en BASIC:

```
FOR X = 1 TO 10
PRINT "X = ";X
FOR Y = 1 TO 10
GOSUB SCAN
NEXT Y
PRINT "C$ = ";C$
NEXT X
```

Éste es un ejemplo de bucles FOR...NEXT anidados. Cuando el intérprete de BASIC llega a la segunda sentencia FOR, necesita recordar cuál es la variable utilizada para el FOR anterior (X, en este caso) y por tanto "empuja" la información relativa al primer FOR dentro de una "pila". Cuando se ha completado el bucle interior, hace "saltar" la información desde el extremo superior de la pila y sabe que el FOR actual utiliza la variable X. Dado que los bucles FOR...NEXT se pueden anidar tan profundamente como sea necesario, podría necesitar empujar en la "pila" información para varios FOR. Cuando hace saltar la información de la "pila", obviamente necesita tener la información en orden inverso al orden en que fue empujada.

Por el contrario, los buffers organizan la memoria de modo que la primera información que entra es la primera información que sale. Los buffers se suelen utilizar para rutinas de entrada/salida y se emplean como "interfaces" entre rutinas o dispositivos que trabajan en unidades diferentes o a distintas velocidades. Por ejemplo, una rutina de entrada en BASIC podría trabajar en unidades de líneas, terminadas por un Carriage Return <CR>, pero el intérprete podría funcionar en las líneas por unidades de un carácter. Por lo general los buffers necesitan de un "señalador" que indique en qué lugar del buffer se ha de escribir el siguiente carácter. El señalador serían uno o varios bytes que contuvieran la dirección de aquel carácter. La dirección se incrementaría después de que se hubiera almacenado cada carácter.

Señales poderosas
La lógica interna del ordenador funciona a niveles TTL. Transistor-Transistor-Logic significa un 1 binario con cinco voltios y un 0 con cero voltios. No obstante, aunque dispositivos como la CPU pueden producir estos voltajes, no pueden generar corriente suficiente para activar todos los otros chips que podrían estar conectados a cada patilla. Por tanto, se conectan buffers de señal a las líneas de salida de la CPU para aumentar la cantidad de corriente. Los buffers de señal son pequeños chips, y cada uno de ellos actúa como un buffer para seis señales

Cálculo analógico

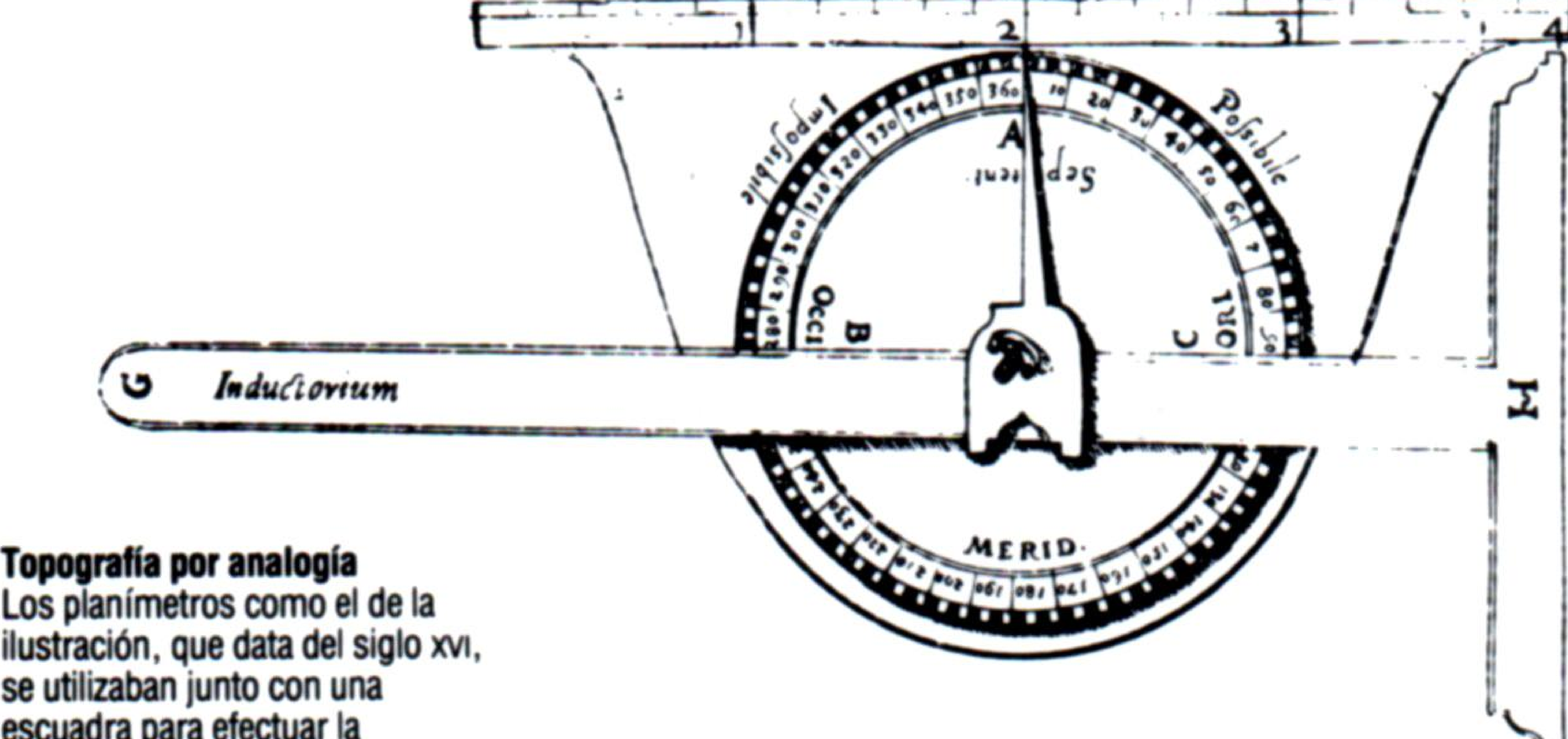
Los ordenadores analógicos, utilizados para controlar máquinas y procesos, reaccionan directamente a los cambios del mundo real, sin que sea preciso traducir la información a forma digital



Cortesía de Physical & Electronics Labs Ltd

Cuadro de mandos
Los ordenadores analógicos no utilizan lenguajes como el BASIC. Se "programan" empalmando diversos componentes eléctricos. Los componentes se fijan a la cara posterior de un tablero de circuitos. En la cara anterior se colocan los enchufes y conectores que permiten conectar entre sí los componentes seleccionados

Existen dos familias de ordenadores bastante diferentes, y hasta ahora sólo nos hemos estado ocupando de una de ellas: el ordenador digital, así llamado porque todas las instrucciones de un programa y todos los datos se representan utilizando "dígitos" binarios. La otra familia, la más antigua de las dos, es la de los ordenadores analógicos, que se programan de forma diferente: uniendo varios componentes electrónicos.



Topografía por analogía
Los planímetros como el de la ilustración, que data del siglo XVI, se utilizaban junto con una escuadra para efectuar la triangulación *in situ*. Anteriormente, los topógrafos tenían que tomar medidas y luego efectuar cálculos matemáticos para determinar la longitud del lado más lejano de su triángulo, pero gracias a este sencillo instrumento analógico su trabajo se volvió mucho más sencillo y exacto. Aún hoy en día se emplean instrumentos de nombre similar para medir la superficie de planos irregulares, como pueden ser las pieles de animales

El velocímetro constituye un ejemplo de este ordenador "analógico", especializado y sencillo; el nombre proviene del comportamiento "análogo" de la velocidad del coche respecto a la posición de la aguja en el dial, siendo aquélla directamente proporcional a ésta. Los ordenadores analógicos modernos pueden realizar muchas tareas y se basan en el tipo de componentes eléctricos que encontramos comúnmente en los aparatos domésticos: transistores, condensadores, resistencias e inductancias magnéticas. Cuando se comenzaba a desarrollar la electrónica, se descubrió que el comportamiento de los componentes eléctricos se parecía al de los dispositivos mecánicos. Por ejemplo, los ingenieros eléctricos descubrieron que las oscilaciones de corriente eléctrica que podían resultar al conectar entre sí una inductancia magnética y un condensador se asemejaban muchísimo a las oscilaciones de una pesa colgada de una cuerda. De hecho, la descripción matemática de ambos sistemas era idéntica: la base para el "cálculo analógico".

Algunos dispositivos analógicos se asemajan mucho a los sistemas que ejemplifican; por ejemplo, el modelo de un aeroplano que se emplea para un experimento de túnel de viento es una copia exacta, a escala reducida, de la armazón del avión. Otros modelos resultan muy distintos. Por ejemplo, un "modelo" de una situación de la vida real puede consistir tan sólo en una lista de fórmulas matemáticas, o puede ser un circuito eléctrico que reproduce el flujo de agua a través de una presa.

Las máquinas de calcular que utilizan principios analógicos demostraron por primera vez su importancia en 1630, cuando William Oughtred inventó la regla de cálculo. Los números estaban dispuestos sobre dos reglas de modo tal (estaban espaciados logarítmicamente) que el movimiento de una a lo largo de la otra equivalía a la multiplicación, y la respuesta simplemente se podía leer de la escala.

A fines del siglo XIX lord Kelvin diseñó un ingenioso dispositivo mecánico a manivela que se podía emplear para calcular las mareas altas y bajas de un puerto durante todo el año.

Luego, en 1930, se construyó en Estados Unidos una máquina electromecánica que podía resolver ecuaciones diferenciales generales (el tipo de ecuaciones que se descubre prácticamente en cualquier representación matemática del mundo real), en vez de la serie específica de ecuaciones diferenciales que Kelvin había conseguido resolver. La máquina la inventó Vannevar Bush, quien le dio la denominación de analizador diferencial.

Los primeros ordenadores analógicos

Los primeros ordenadores analógicos totalmente electrónicos entraron en funcionamiento en 1947, justo después de que nacieran los primeros ordenadores digitales. Hemos visto cómo los ordenadores digitales realizan cálculos aritméticos utilizando una combinación de puertas lógicas (véase p. 68). El ordenador analógico puede efectuar cálculos matemáticos simplemente utilizando la naturaleza de la electricidad. Si hay, por ejemplo, una corriente eléctrica de cinco amperios que fluye por un cable y una corriente de cuatro amperios que fluye por otro, y los dos cables están

unidos entre sí para converger en un único cable, la corriente del nuevo cable será de nueve amperios: la suma de las dos corrientes. De manera, por lo tanto, que el conjunto del comportamiento y las propiedades del flujo de corriente constituyen automáticamente un "sumador".

Hasta para las funciones matemáticas muy complicadas a menudo se hallan soluciones muy sencillas en circuitos elementales (un ejemplo es la "integración": hallar la superficie bajo una curva). Los ordenadores analógicos no se "programan" como las máquinas digitales; en cambio, se ha de construir un circuito que ejemplifique el programa a resolver. Todos los componentes disponibles se montan en la parte posterior de un "tablero", en cuya cara anterior están los enchufes y conectores para unir los componentes entre sí mediante cables. El aspecto de este "cuadro de mandos" recuerda mucho a un anticuado cuadro conmutador de teléfonos.

En un ordenador analógico se utilizan los diversos voltajes o corrientes para representar cantidades físicas como fuerza o velocidad, y los "valores" de los componentes eléctricos representan cosas como la masa de un coche o la fuerza de sus amortiguadores. Pero en un ordenador digital todos los datos se representan mediante series de impulsos, cinco voltios para el 1 binario y cero voltios para el 0 binario. Aquí hallamos una distinción más amplia entre los ordenadores analógicos y los digitales: en unos la información se puede almacenar de forma que satisfaga cantidades que cambian continuamente, pero en los otros los datos se almacenan en unidades "discretas" o individuales.

Algunos ordenadores personales (el BBC Micro, por ejemplo) poseen un conector para entrada analógica. En general, cuando los ordenadores digitales manipulan información analógica, como una temperatura o una fuerza, primero han de convertir los datos en forma digital.

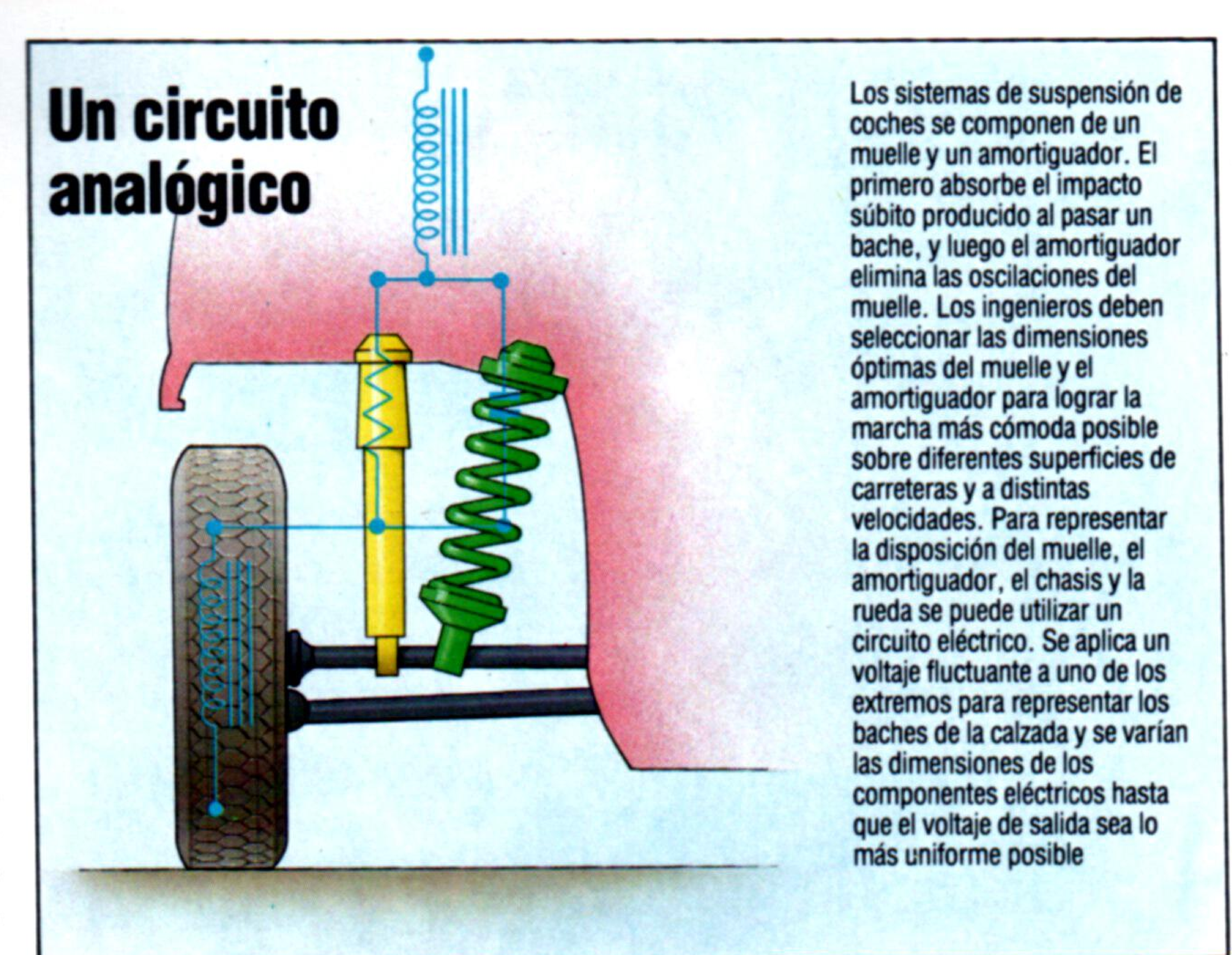
La gran ventaja de los sistemas digitales es que la información se puede procesar o transmitir sin pérdida de calidad. Si un impulso de cinco voltios se hace pasar a través de un circuito eléctrico, podría muy bien ser afectado por la distorsión inherente a todo circuito y salir, quizá, como 4,9 voltios. En un sistema analógico, en el cual las fluctuaciones de voltaje representan cambios en la información, esto podría significar, en la voz de un cantante, la diferencia que existe entre un la y un la bemol. Pero en un sistema digital, en el cual sólo hay dos señales posibles, cinco voltios o cero, cualquier señal próxima a cinco (por ejemplo, 4,9) se reconoce y se reproduce automáticamente como cinco voltios. De modo que los errores se corrigen y no se acumulan. Por el contrario, la acumulación de errores a medida que la señal pasa a través de circuitos sucesivos representa uno de los inconvenientes del ordenador analógico.

No obstante, este tipo de ordenadores saca provecho de representar las cantidades como valores variables de un voltaje o una corriente. Esto significa que una condición de entrada se puede cambiar rápidamente y que el sistema reflejará de inmediato las modificaciones consiguientes. No se necesita tiempo para codificar los datos en impulsos binarios, ni para procesarlos ni para, finalmente, volver a decodificarlos para producir la salida.

Esta característica es muy importante para las aplicaciones en las que resulta esencial producir respuestas rápidas. Por ejemplo, un piloto automático debe responder a una repentina corriente de viento durante un aterrizaje, cuando ya no hay tiempo para efectuar cálculos largos, ni siquiera a la velocidad a que funcionan los ordenadores digitales modernos. Los sensores detectan la corriente repentina, generando un voltaje de salida relativamente pequeño. El circuito del piloto automático responde de manera instantánea con un cambio del voltaje de salida relativamente grande, que de forma automática activa los alerones de las alas, de tal manera que el aparato consiga mantener la estabilidad sin mayores percances.

Los ordenadores analógicos se utilizan en muchas áreas del control industrial, en las cuales se han de manejar con sumo cuidado complicados sistemas mediante ajustes sutiles y continuos, como en una planta industrial química. Pero son menos conocidos que sus equivalentes digitales. Aunque algunas veces se emplean ordenadores analógicos sencillos en las escuelas, con fines pedagógicos, el tipo de aplicaciones para las cuales resultan idóneos revela que es muy poco probable que lleguemos alguna vez a ver un ordenador personal analógico. Los ordenadores analógicos se utilizarán siempre, pero serán los ordenadores digitales los que irán dominando el mercado a medida que vayan siendo más rápidos y más potentes.

Un circuito analógico



Los sistemas de suspensión de coches se componen de un muelle y un amortiguador. El primero absorbe el impacto súbito producido al pasar un bache, y luego el amortiguador elimina las oscilaciones del muelle. Los ingenieros deben seleccionar las dimensiones óptimas del muelle y el amortiguador para lograr la marcha más cómoda posible sobre diferentes superficies de carreteras y a distintas velocidades. Para representar la disposición del muelle, el amortiguador, el chasis y la rueda se puede utilizar un circuito eléctrico. Se aplica un voltaje fluctuante a uno de los extremos para representar los baches de la calzada y se varían las dimensiones de los componentes eléctricos hasta que el voltaje de salida sea lo más uniforme posible

Kevin Jones

Mensaje distorsionado



La manera en que se va alterando un mensaje cuando se transmite de oído a oído tiene gran similitud con los errores acumulativos de un circuito analógico

En un circuito digital sólo se puede pasar un número limitado de mensajes (concretamente 1 o 0). De modo que si se produjera cualquier distorsión, ésta se podría identificar fácilmente y eliminar en la siguiente etapa

Herman Hollerith

1860
Nace en Buffalo (estado de Nueva York)

1879
Se gradúa en la Universidad de Columbia y consigue empleo en la US National Census Office

1882
Se matricula en el Massachusetts Institute of Technology para realizar investigación

1883
Trabaja en Washington para la Patent Office

1884
Solicita sus primeras patentes para codificar información sobre una cinta continua de papel perforado

1887
Se adopta su sistema para procesar las estadísticas relativas a la mortalidad en Baltimore (Maryland)

1889
Se instala un sistema en la oficina de los jefes militares de sanidad para organizar las estadísticas médicas del ejército. Registra una patente por el invento de la tarjeta perforada individual

1890
Gana el concurso para provisión de equipos para procesar el censo de 1890. Obtiene el Doctorado en Filosofía por la Universidad de Columbia por su trabajo en el campo del procesamiento de la información

1900
Presenta una nueva generación de equipos perfeccionados

1901
Se utilizan los nuevos equipos para el censo agrícola

1905
Comienzan a caducar sus primeras patentes y la competencia empieza a luchar contra su monopolio

1911
Funda un holding: la Tabulating Recording Company

1914
El famoso administrador de empresas Thomas J. Watson asume la dirección del holding

1924
La empresa cambia de nombre y pasa a denominarse International Business Machines (IBM)

1929
Muere en Washington, D.C.


Cortesía de IBM Uk Ltd

El inventor que colocó las cifras de la población de Estados Unidos en tarjetas perforadas y fundó la mayor empresa de informática del mundo

Hollerith nació en Estados Unidos en 1860. Después de graduarse en la Universidad de Columbia consiguió empleo en la US National Census Office (Oficina Nacional de Empadronamiento), ayudando a elaborar estadísticas tomadas del censo de 1880. Todo el trabajo se realizaba a mano y resultaba concienzudamente lento; lo era tanto que, de hecho, cuando diez años más tarde llegó el momento de realizar el censo siguiente, en la oficina todavía no habían terminado de tabular los resultados del anterior. Hollerith sabía que su mayor aptitud era su poder de invención; con el objeto de entrenarse como inventor y desarrollar sus cualidades creativas, abandonó su trabajo en la National Census Office e ingresó en la Patent Office (Oficina de Patentes) de Washington.

La primera idea de Hollerith fue la de codificar información en cinta de papel. La cinta de papel estaba dividida en "campos" mediante marcas de tinta. Cada campo representaba diferentes categorías: por ejemplo, varón o mujer o blanco o negro. La presencia de un agujero en el campo varón/mujer significaba que la persona era un varón, mientras que su ausencia implicaba que se trataba de una mujer, y así sucesivamente. Estos agujeros después se podían "leer" mediante una máquina. Sus primeras patentes se produjeron en 1884 y dedicó los años siguientes a perfeccionar el sistema. Empezó por procesar la información relativa a las estadísticas de sanidad de las ciudades norteamericanas, que se estaban desarrollando rápidamente, y de la administración militar.

Cinco años después, en 1889, perfeccionó la idea de la cinta de papel perforado utilizando tarjetas separadas para cada persona. Las tarjetas eran del tamaño de los billetes de un dólar; se dice que esto se debió, en parte, a que los únicos equipos que se pudieron adaptar habían sido construidos para manipular dinero. Originalmente los agujeros eran redondos y se hacían con el punzón que utilizaban los conductores de autobuses para perforar los billetes, pero luego se construyeron punzones especiales para cortar un agujero cuadrado de 6 mm. Así se podía incluir gran cantidad de información en una sola tarjeta.

La ventaja que ofrecen las tarjetas individuales sobre la cinta continua es que, además de obtener los totales, la información se puede clasificar. Por ejemplo, a usted tal vez le interese saber qué cantidad de mujeres blancas de 80 años de edad viven en la ciudad de Nueva York. Se podrían clasificar todas las tarjetas y las que tuvieran agujeros perforados en estos tres campos se podrían separar mecánicamente del resto. Estas primeras máquinas sólo podían producir totales, pero posteriormente Hollerith introdujo la suma y otras operaciones aritméticas sencillas.

El éxito comercial llegó en 1889, cuando el Bureau of Censuses (Departamento de Censos) convocó un concurso para proveerse de un sistema de equipos que procesara el censo que se realizaría al año siguiente. Los sistemas se probaron volviendo a tabular los datos obtenidos del censo anterior. La convocatoria la ganó el equipo de Hollerith. Para entonces todas sus máquinas estaban amparadas bajo patentes y él aprovechó su monopolio para cargarle al gobierno 65 céntimos por cada mil tarjetas procesadas. Aunque cada habitante de Estados Unidos tenía su propia tarjeta individual, Hollerith sólo tardó dos años en hacer el trabajo. Anunció que la población del país era de 56 millones de habitantes y presentó la factura al gobierno.

Cuando llegó el momento de realizar el censo de 1900, Hollerith había desarrollado una maquinaria muchísimo más eficaz, pero se negó a rebajar su tarifa. Cuando caducaron sus patentes, el gobierno buscó otras empresas, pero Hollerith superó a la competencia fundando su propia compañía, que posteriormente se convertiría en la IBM (International Business Machines), firma que hoy está a la cabeza de las empresas fabricantes de ordenadores en todo el mundo.

Juegos de tarjetas
El método original de Hollerith para representar la información se sigue utilizando en la actualidad, un siglo después, si bien se ha modificado el formato de la tarjeta. Las modernas tarjetas perforadas constan de 12 filas y 80 columnas. Las máquinas tabuladoras utilizaban el sistema decimal y, por tanto, cada ficha podía almacenar 80 números. Los caracteres alfabéticos se crearon con la "multiperforación": efectuando más de un agujero en una columna. Los ordenadores también aceptan fichas perforadas en sistema binario



Ya están a su disposición, en todos los quioscos y librerías, las tapas intercambiables para encuadernar 12 fascículos de

mi computer

Cada juego de tapas va acompañado de una colección de transferibles, para que usted mismo pueda colocar en cada lomo el número de tomo que corresponda





10012
9 788485 822836